Course Title:

System Analysis and Design (BIT 253)

BIT 4th semester-Tribhuvan University (TU)

Compiled by Ujjwal Rijal
rijalujjwal09@gmail.com

rijalujjwal09@gmail.com

Unit-1

Foundations for Systems Development

Compiled by Ujjwal Rijal
rijalujjwal09@gmail.com

rijalujjwal09@gmail.com

The Systems Development Environment

System

- The word **System** is derived from Greek word **Systema**, which means an organized relationship between any set of components to achieve some common cause or objective.
- A system is a set of components that interact to accomplish some purposes. For example: language system, economic system, college system, marketing system, accounting, shipping, Government and so on.
- A system is "an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal."

rijalujjwal09@gmail.com

Contd. ….

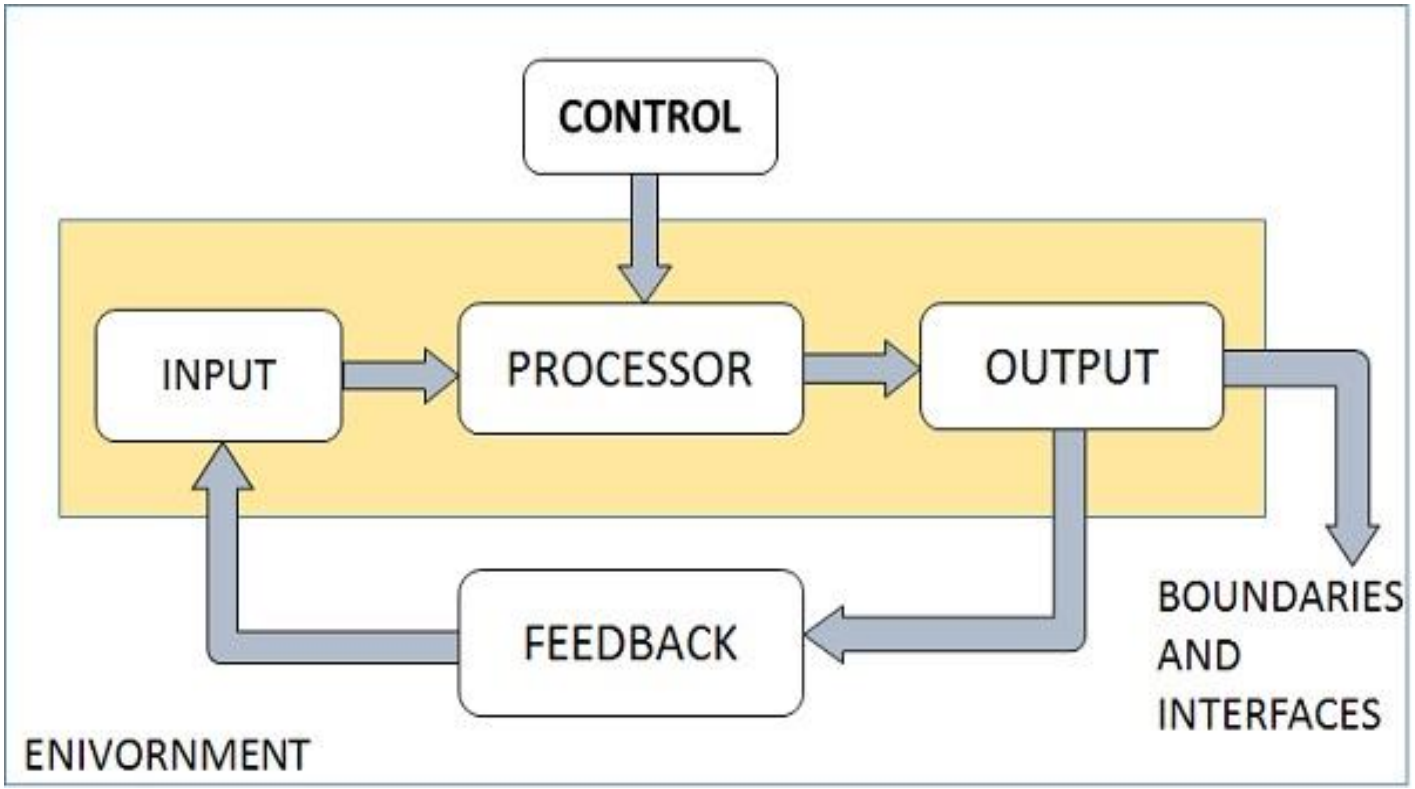


Fig: Elements/ Components of System

rijalujjwal09@gmail.com

Contd. ….

- **Outputs and Inputs**

✓ The main aim of a system is to produce an output which is useful for its user.

✓ Inputs are the information that enters into the system for processing.

✓ Output is the outcome of processing.

- **Processors**

✓ The processor is the element of a system that involves the actual transformation of input into output.

✓ It is the operational component of a system. Processors may modify the input either totally or partially, depending on the output specification.

✓ As the output specifications change, so does the processing. In some cases, input is also modified to enable the processor for handling the transformation.

rijalujjwal09@gmail.com

Contd. ….

- **Control**

✓ The control element guides the system.

✓ It is the decision–making subsystem that controls the pattern of activities governing input, processing, and output.

✓ The behavior of a computer System is controlled by the Operating System and software. In order to keep system in balance, what and how much input is needed is determined by Output Specifications.

- **Feedback**

✓ Feedback provides the control in a dynamic system.

✓ Positive feedback is routine in nature that encourages the performance of the system.

✓ Negative feedback is informational in nature that provides the controller with information for action.

rijalujjwal09@gmail.com

Contd. ….

- **Environment**

✓ The environment is the "super system" within which an organization operates.

✓ It is the source of external elements that strike on the system.

✓ It determines how a system must function. For example, vendors and competitors of organization's environment, may provide constraints that affect the actual performance of the business.

- **Boundaries and Interfaces**

✓ A system should be defined by its boundaries. Boundaries are the limits that identify its components, processes, and interrelationship when it interfaces with another system.

✓ Each system has boundaries that determine its sphere of influence and control.

✓ The knowledge of the boundaries of a given system is crucial in determining the nature of its interface with other systems for successful design.

rijalujjwal09@gmail.com

Contd. ….

❖ Characteristics/ Properties of a System

- **Organization**

✓ Organization implies structure and order. It is the arrangement of components that helps to achieve predetermined objectives.

- **Interaction**

✓ It is defined by the manner in which the components operate with each other.

✓ For example, in an organization, purchasing department must interact with production department and payroll with personnel department.

- **Interdependence**

✓ Interdependence means how the components of a system depend on one another. For proper functioning, the components are coordinated and linked together according to a specified plan.

✓ The output of one subsystem is required by other subsystem as input.

rijalujjwal09@gmail.com

Contd. ….

- **Integration**

✓ Integration is concerned with how a system components are connected together. It means that the parts of the system work together within the system even if each part performs a unique function.

- **Central Objective**

✓ The objective of system must be central. It may be real or stated. It is not uncommon for an organization to state an objective and operate to achieve another.

✓ The users must know the main objective of a computer application early in the analysis for a successful design and conversion.

rijalujjwal09@gmail.com

Contd. ….

❖ Information System (IS)

- It is interrelated component working together to collect, process, store, and disseminate information in order to support decision making, coordination control analysis and visualization in an organization.
- Moreover, the system analysis and design (SAD) is the process of developing information system (IS) that effectively uses hardware, software, data, process and people to support company's unique business objectives.
- ***The basic information system model is shown in the below figure:***

rijalujjwal09@gmail.com

Contd. ….

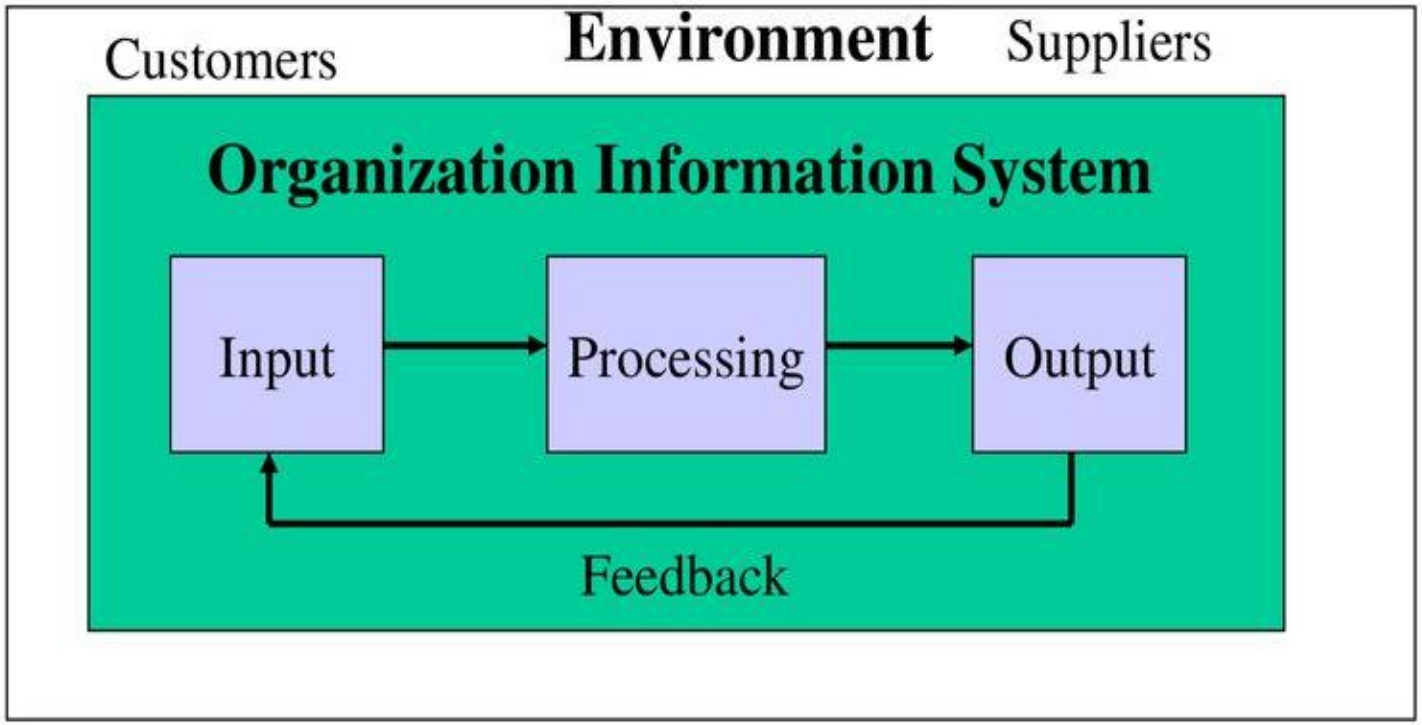


Fig: Basic Information System Model

rijalujjwal09@gmail.com

Contd. ….

➢ Typical Components of Information System

- **Hardware:** Computer-based information systems use computer hardware, such as processors, monitors, keyboard and printers.
- **Software:** These are the programs used to organize, process, and analyze the data.
- **Databases:** Information systems work with data, organized into tables and files.
- **Network:** Different elements need to be connected to each other, especially if many different people in an organization use the same information system.
- **Procedures:** These describe how specific data are processed and analyzed in order to get the answers for which the information system is designed.

rijalujjwal09@gmail.com

Contd. ….

➢ Types of Information System

- There are various types of computer-based ISs, which differ in their business needs and also depend upon different levels in the organizational ISs.
- ***Some of the information systems are discussed below:***

• Office Automation System (OAS)

✓ OAS is a collection of software and hardware products that increase productivity within the office setting. It can be described as a multi-function, integrated computer based system that allows many office activities to be performed in an electronic mode.

✓ OASs are among the newest and most rapidly expanding computer based information system. They are being developed with the hopes and expectations that they will increase efficiency and productivity of office worker, typist, administrative assistance, staff professionals, managers, etc.

✓ Many organizations have taken the first step towards automating their office.

rijalujjwal09@gmail.com

Contd. ….

✓ **The activities supported by OAS include:**

- Scheduling resources: **Examples**: Electronic calendars with resource management (equipment, facilities, etc.)
- Document preparation: **Examples**: Software (Word processing and desktop publishing), hardware (printers).
- Communication: **Examples**: Email, voicemail, chat, video conferencing, etc.

Advantages of OAS

- Office automation can get many tasks accomplished faster, eliminating the need for a larger set of staffs.
- Less storage space is required for data and copies can be easily transferred off-site for safe-keeping in case of fire and other emergencies.
- Multiple people can be updated simultaneously in the event of schedule changes, etc.

Disadvantages of OAS

- Older staff members may have a harder time adjusting to the new technology and be unable to use it efficiently and effectively.
- If something is misfiled, it can be harder to find.
- The amount of money required to implement and the cost of maintenance of certain equipment is extremely larger, etc.

rijalujjwal09@gmail.com

Contd. ….

- ## Transaction Processing System (TPS)

✓ TPS is a special class of information system designed to process business events and transactions.

✓ **There are two types of TPS: Batch transaction processing** and **Real time processing**.

✓ Batch transaction processing collects the transaction data as a group or batch, and processes it later.

✓ Whereas, Real time processing is the immediate processing of data. It provides instant confirmation of transaction but requires access to online base.

✓ **Examples of TPS include**: Pay cheques and other forms of paper output, airline reservation system, banking system, purchase order entry system, etc.

rijalujjwal09@gmail.com

Contd. ….

Advantages of TPS

- Handlings of several thousands of operations at once.
- In TPS, there is no delay and the results of each transaction are immediately available, etc.

Disadvantages of TPS

- Needs to handle hundreds, even thousands of simultaneous users.
- Needs to allow many results to work on the some set of data, with immediate updating, etc.

rijalujjwal09@gmail.com

Contd. ….

- **Management Information System (MIS)**

✓ MIS is used by the management employees to support recurring decision making in managing a function or the entire business.

✓ These systems assist lower management in problem solving and making decision. They use the result of transaction processing and some other information also.

✓ It is a set of information processing function. It should handle queries as quickly as they arrive. An important element of MIS is database.

✓ **Examples**: Annual budgeting, sales order system, personnel management system, etc.

rijalujjwal09@gmail.com

Contd. ....

Advantages of MIS

- It facilitates planning.
- It minimizes information overload by changing the larger amount of data in order to summarize its form.
- It encourages decentralization by making necessary changes in the organizational plans and procedures.
- It brings co-ordination as it connects all decision centres in the organization, etc.

Disadvantages of MIS

- Highly sensitive; requires constant monitoring.
- Budgeting of MIS is extremely high and difficult.
- Quality of output is governed by the quality of inputs.
- Effectiveness decreases due to the frequent changes in the top level management, etc.

rijalujjwal09@gmail.com

Contd. ….

- **Decision Support System (DSS)**

✓ DSS is a special purpose information system designed to support managerial level employees in organizational decision making.

✓ DSS is an organized collection of people, procedures, database and devices that are used to support problem specific decision making.

✓ These systems assist top level management to make long term decisions. These types of system handle unstructured and semi-structured decisions.

✓ A decision is considered unstructured if there is no clear procedure for making the decision and if not all the factors to be considered in the decision can be readily identified in advance.

rijalujjwal09@gmail.com

Contd. ….

Advantages of DSS

- Reducing decision cycle time, increased employee productivity and more timely information is obtained for decision making.
- Improved decision making, effectiveness and better decision making.
- Improved communication and collaboration among decision makers.
- Gaining of competitive advantages from computerized decision support.
- Promote learning and increased organizational control, etc.

Disadvantages of DSS

- It requires heavy investment in information system to collect data from many sources and analyse them to support the decision making.
- It is conceivable and has been demonstrative that some DSS reduce the skill needed to perform a decision task.
- The computer does not make a bad decision, people do. Unfortunately, some people may deflect personal responsibility to a DSS, etc.

rijalujjwal09@gmail.com

Contd. ….

- **Expert System (ES)**

✓ ES is a special purpose system used by top level employees to make decision, usually made by more experienced employees or an expert in the field.

✓ ES is a programmed decision making information system and reproduces the knowledge and expertise of an expert problem solver, managers, technicians, etc.

✓ ES replicates decision making process and is actually a knowledge based system that describes the way an expert would approach the problems.

✓ ES is an information system that can function as a consultant to a problem solver, not only suggesting to a solution but also by explaining the line of reassuring that leads to the solution as a what a human expert can do.

✓ ES imitates the logic and reasoning of the expert within their respective fields. ES is the branch of AI and is also called knowledge-base (KB) system.

rijalujjwal09@gmail.com

Contd. ….

Advantages of ES

- Provides consistent answer for repetitive decisions, processes and tasks.
- Holds and maintains significant levels of information.
- Encourages organization to clarify the logic of their decision makings.
- Saves the human efforts and time, and can work round the clock.
- It makes scarce expertise more widely available, etc.

Disadvantages of ES

- Can not make creative response as human expert would make in unusual circumstances.
- Domain experts are not always able to explain their logic and reasoning.
- Errors may occur in the KB, and may lead to the wrong decisions.
- Can not adapt to the changing or dynamic environment unless KB is changed, etc.

rijalujjwal09@gmail.com

Contd. ….

- **Executive Support System (ESS)**

- ✓ ESS, also called as Executive Information System (EIS) is a special purpose information system that supports executives in decision making. It is primarily used by the top-level management.
- ✓ It is a user-friendly, interactive system and almost intuitive to use; it has excellent menus and graphic capabilities designed to meet the information needs of top level management engaged in long-range planning, crisis management, and other strategic decisions.
- ✓ Such systems assist in the making of decisions that require an in-depth understanding of the firm and of the industry in which the firm operates.
- ✓ Another special characteristic of an ESS is its drill-down capability, which is the ability of the system to provide information of any levels of detail-design by the decision makers.
- ✓ The activities supported by these kinds of system include; Executive decision making, long range strategic planning, monitoring of internal and external events, crisis management, staffing and labour relations management, etc.

rijalujjwal09@gmail.com

Contd. ….

Advantages of ESS

- Filters data for management.
- Improves towards tracking information.
- Offers efficiency to the decision makers.
- Easy for upper-level executives to use; extensive computer experience is not required during the process of operations.
- Provides timely delivery of company's summarized information, etc.

Disadvantages of ESS

- Difficult to keep current data.
- May lead to less reliable and insecure data.
- Small companies may encounter excessive cost for implementation.
- System dependent and limited functionalities through design.
- Information overload for some managers, etc.

rijalujjwal09@gmail.com

❑ Overview of System Analysis and Design

- **System analysis and design (SAD)** is the complex, challenging, and simulating organizational process that a team of business and systems professionals uses to develop and maintain computer based information systems.
- **System Analysis** is the process of gathering and interpreting facts, diagnosing problems, and using the facts to improve the system. **Analysis** specifies what the system should do.
- **System Design** is the process of planning a new system to replace or complement the old. **Design** states how to achieve the objective.

rijalujjwal09@gmail.com

Contd. ….

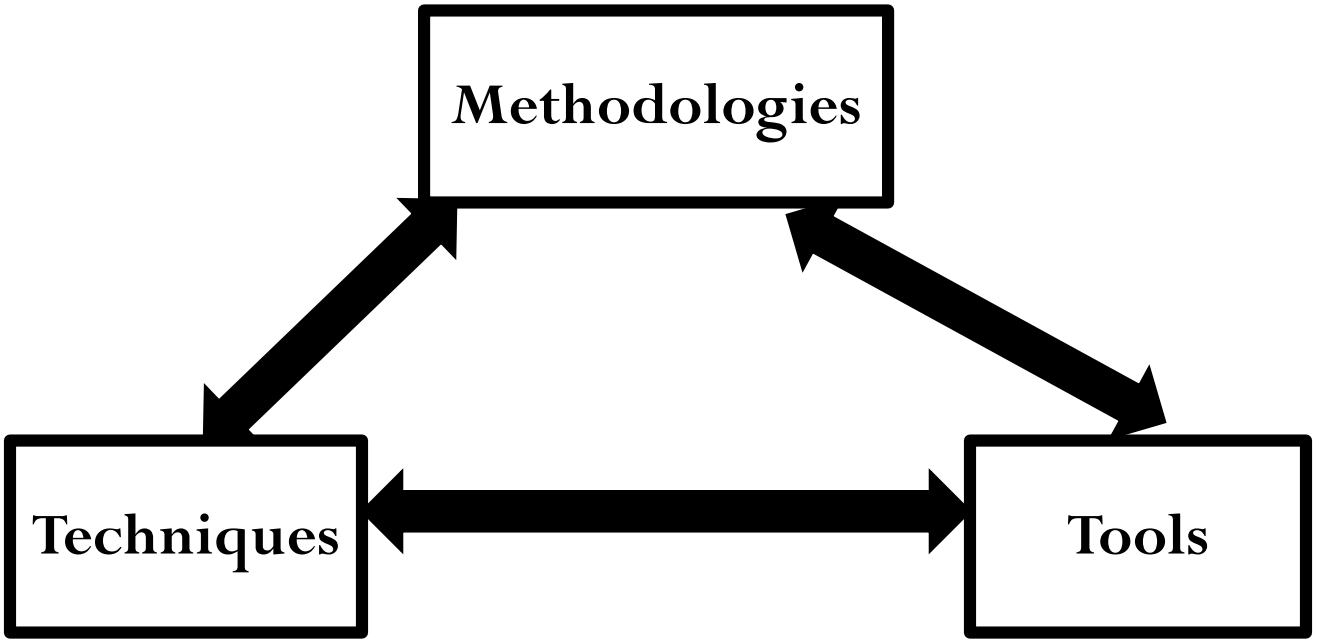


Fig: An organizational approach to systems analysis and design is driven by methodologies, techniques, and tools.

rijalujjwal09@gmail.com

Contd. ….

- **Methodologies** define the set of steps that will guide our work and influence the quality of our final product: the information system. Most methodologies incorporate several development techniques.
- **Techniques** are particular process that will help to ensure that the work is well thought-out, complete, and comprehensible to other on the project team.
- **Techniques** also provide support for a wide range of tasks like conducting interviews, planning and managing the activities in a system development project, diagramming the system's logic, and designing the reports that the system will generate.
- **Tools** are typically computer programs that make it easy to use and benefit from the techniques and to faithfully follow the guidelines of the overall development **methodology**.
- To be effective, both **techniques** and **tools** must be consistent with an organizations' system development **methodology**. These make easy for system developers to conduct the steps in **methodology**.
- **Techniques** and **tools** must make it easy for systems developers to conduct the steps called for in the **methodology**.

rijalujjwal09@gmail.com

❑ A Modern Approach to Systems Analysis and Design

- The growth of computer-based information systems analysis and design methodologies started during the year 1950 to 1960.
- The researchers argued that the software crisis was due to the lack of discipline of programmers and some believed that if formal engineering methodologies would be applied to software development, then production of software would become as predictable an industry as other branches of engineering and they advocated proving all programs correct using models such as the **Capability Maturity Model (CMM)**.
- In 1986, No Silver Bullet article was published by Fred Brooks which described that no individual technology or practice would ever make an improvement in productivity of software within 10 years. So, they realized the need for developing the software in a structured manner.

rijalujjwal09@gmail.com

Contd. ….

- However, it could also be said that there are, in fact, a range of silver bullets today, including spreadsheet calculators, lightweight methodologies, in-site search engines, customized browsers, integrated design-test coding-editors, database report generators and each issue in software is related with only a small portion of the entire problem which makes the systems analysis and design approaches too complex for finding the complete solution to all the problems.
- The new technologies and practices which were developed after 1970-1990 were primarily focused on solving the software issues like software crisis. The major elements used were software tools, formal methods, well defined processes that use the methodologies like OOP, CASE Tools and Structured Programming approaches.

rijalujjwal09@gmail.com

❑ Developing Information Systems and The System Development Life Cycle

- We use various methodologies, techniques and tools that have been developed, tested and widely used over the years to assist people during system analysis and design.
- When developing information systems, most organizations use a standard of steps called the **systems development life cycle** (SDLC) at the common methodology for systems development.
- **SDLC includes phases such as planning, analysis, design, implementation, and maintenance as shown in the below figure:**

rijalujjwal09@gmail.com

Contd. ....

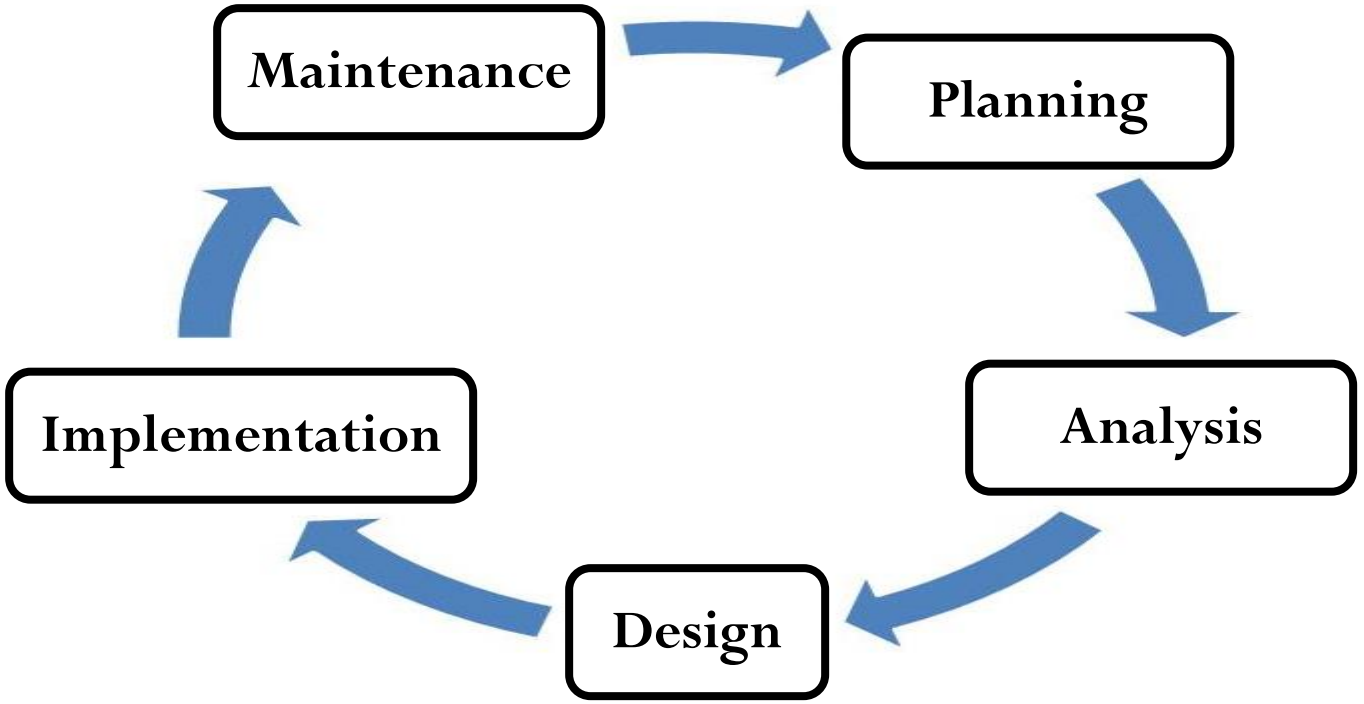


Fig: The System Development Life Cycle

rijalujjwal09@gmail.com

Contd. ….

- The **first phase** is called **planning**. In this phase, an organization's total information system needs are identified, analysed, prioritized, and arranged.
- The **second phase** is called **analysis**. The **analysis** phase usually requires a careful study of the current system, which continues along with **two phases: requirements determination** and **analysis study**.
- **Requirements determination** process usually involves a careful study of the current manual and computerized systems that may be replaced or improved within the project.
- **Analysis study** process usually involves analysts to study the structural requirements according to the components' interrelationships and eliminate the redundancies.

rijalujjwal09@gmail.com

Contd. ….

- The **third phase** is called **design**. In the **design** phase, the description of the recommended solution is converted into **logical** and then **physical** system specification.
- Analysts design all the aspects of the system, provide physical specification on the system from input and output screens to reports, databases, and computer processes.
- **Logical design** is the part of the design process that is independent of any specific hardware or software platform. Theoretically, the system could be implemented on any hardware and system software.
- **Physical design** is the part of the design phase in which the logical specifications of the system form logical design, are transformed into technology-specific details from which all programming and system constructions can be accomplished.

rijalujjwal09@gmail.com

Contd. ….

- The **fourth phase** is called **implementation**. In this phase, the information system is coded, tested, installed, and supported in the organization.
- **During testing**, programmers and analysts test individual programs and the entire system in order to find and correct errors.
- **During installation**, the new system becomes a part of the daily activities of the organization.
- **Implementation activities** also include initial user support such as the finalization of documentation, training programs, and ongoing user assistance in the phase of SDLC.

rijalujjwal09@gmail.com

Contd. ….

- The **fifth and the final phase** is called **maintenance**. In this phase, information system is systematically repaired and improved.
- When a system is operating in an organization, users sometimes find problems with how it works and often think of better ways to perform its functions.
- Also the organization's needs with respect to the system change over time.
- ***The products of SDLC phases are shown in the below table:***

rijalujjwal09@gmail.com

Contd. ….

Table: Products of SDLC Phases

Phase	*Products, Outputs, or Deliverables*
Planning	Priorities for systems and projects; an architecture for data, networks, and selection hardware, and IS management are the result of associated systems; Detailed steps, or work plan, for project; Specification of system scope and planning and high-level system requirements or features; Assignment of team members and other resources; System justification or business case
Analysis	Description of current system and where problems or opportunities are with a general recommendation on how to fix, enhance, or replace current system; Explanation of alternative systems and justification for chosen alternative
Design	Functional, detailed specifications of all system elements (data, processes, inputs, and outputs); Technical, detailed specifications of all system elements (programs, files, network, system software, etc.); Acquisition plan for new technology
Implementation	Code, documentation, training procedures, and support capabilities
Maintenance	New versions or releases of software with associated updates to documentation, training, and support

5

rijalujjwal09@gmail.com

The Heart of the Systems Development Process

- In many ways, though, the SDLC is fiction. Although almost all the systems development projects adhere to some type of life cycle, the exact location of activities and the specific sequencing of steps can vary greatly from one project to the next.
- Current practice combines the activities traditionally thought of as belonging to analysis, design, and implementation into a single process.
- Instead of systems requirements being produced in analysis, systems specifications being created in design, and coding and testing being done at the beginning of implementation, current practice combines all of those activities into a single analysis-design-code-test process as shown in the below figure:

rijalujjwal09@gmail.com

Contd. ….

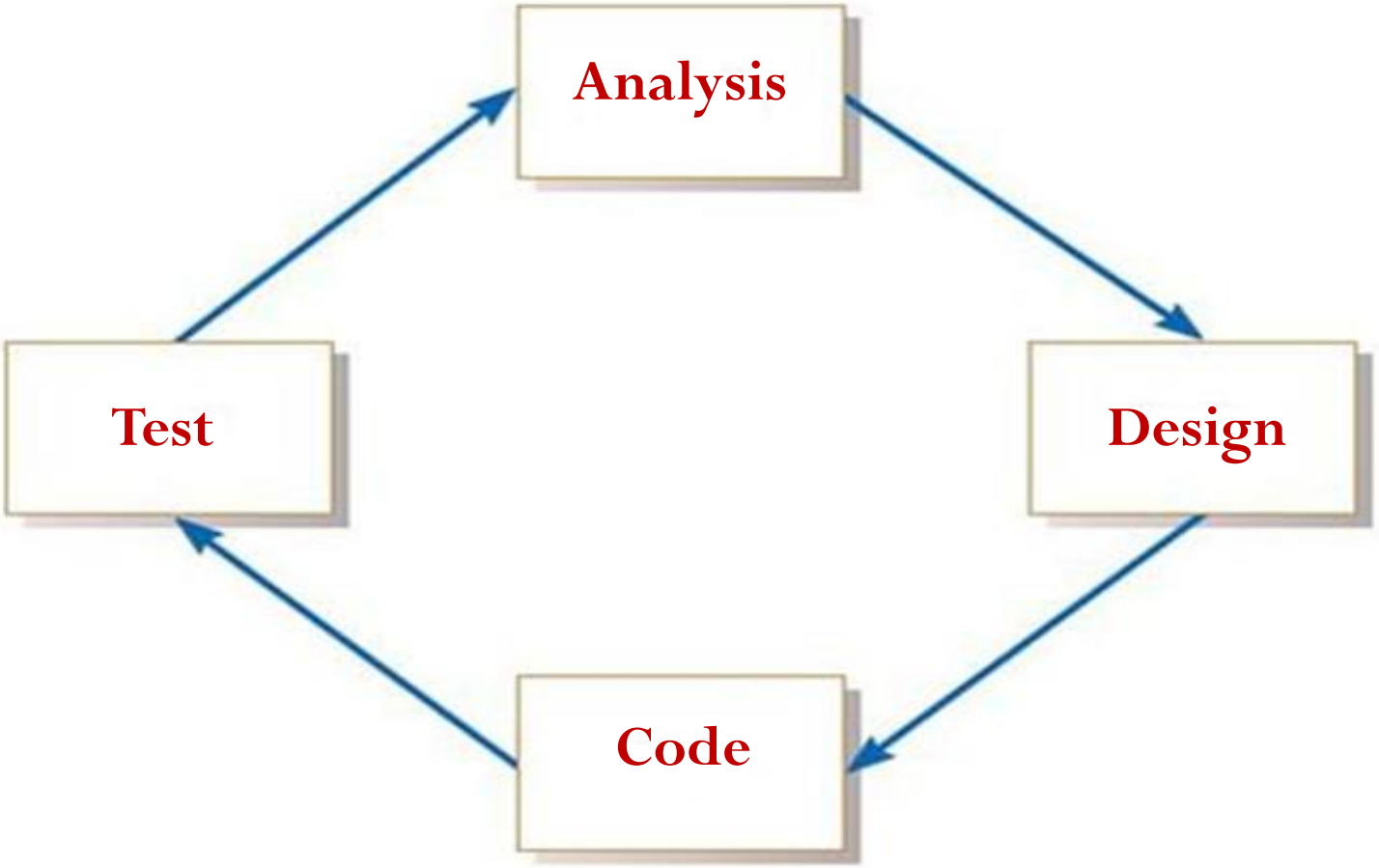


Fig: The analysis-design-code-test loop

rijalujjwal09@gmail.com

Contd. ….

- These activities are the heart of the systems development, as suggested in the below figure. This combination of activities is typical of current practices in Agile Methodologies. A well-known instance of one of the **Agile Methodologies** is **eXtreme Programming**, although there are other variations as well.

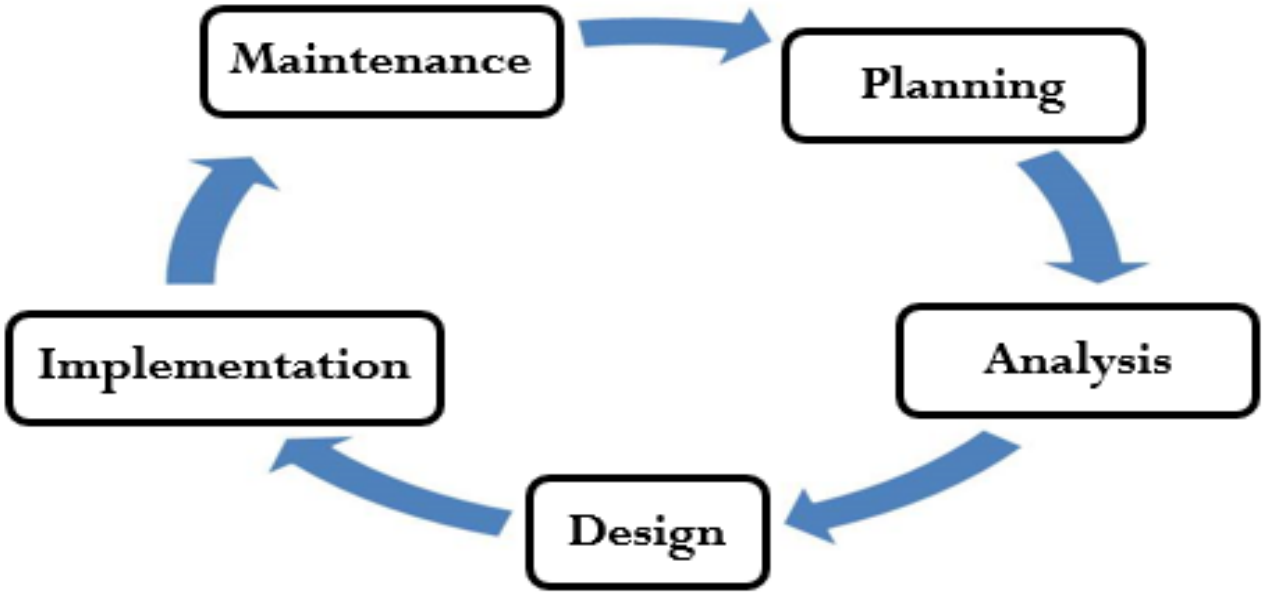


Fig:The heart of the system development

rijalujjwal09@gmail.com

CASE Tools

- **Computer-Aided Systems Engineering (CASE)**, **also called Computer-Aided Software Engineering (CASE)**, is a technique that uses powerful software, called **CASE tools**, in order to help system analysts develop and maintain information systems.
- **CASE tools** provide an overall framework for systems development and support a wide variety of design methodologies, including structured analysis and object-oriented analysis.
- After developing a model, many **CASE tools** can generate program code, which speeds the implementation process.
- ***The following figure shows the sample for Visual Case Tool Software:***

rijalujjwal09@gmail.com

Contd. ...

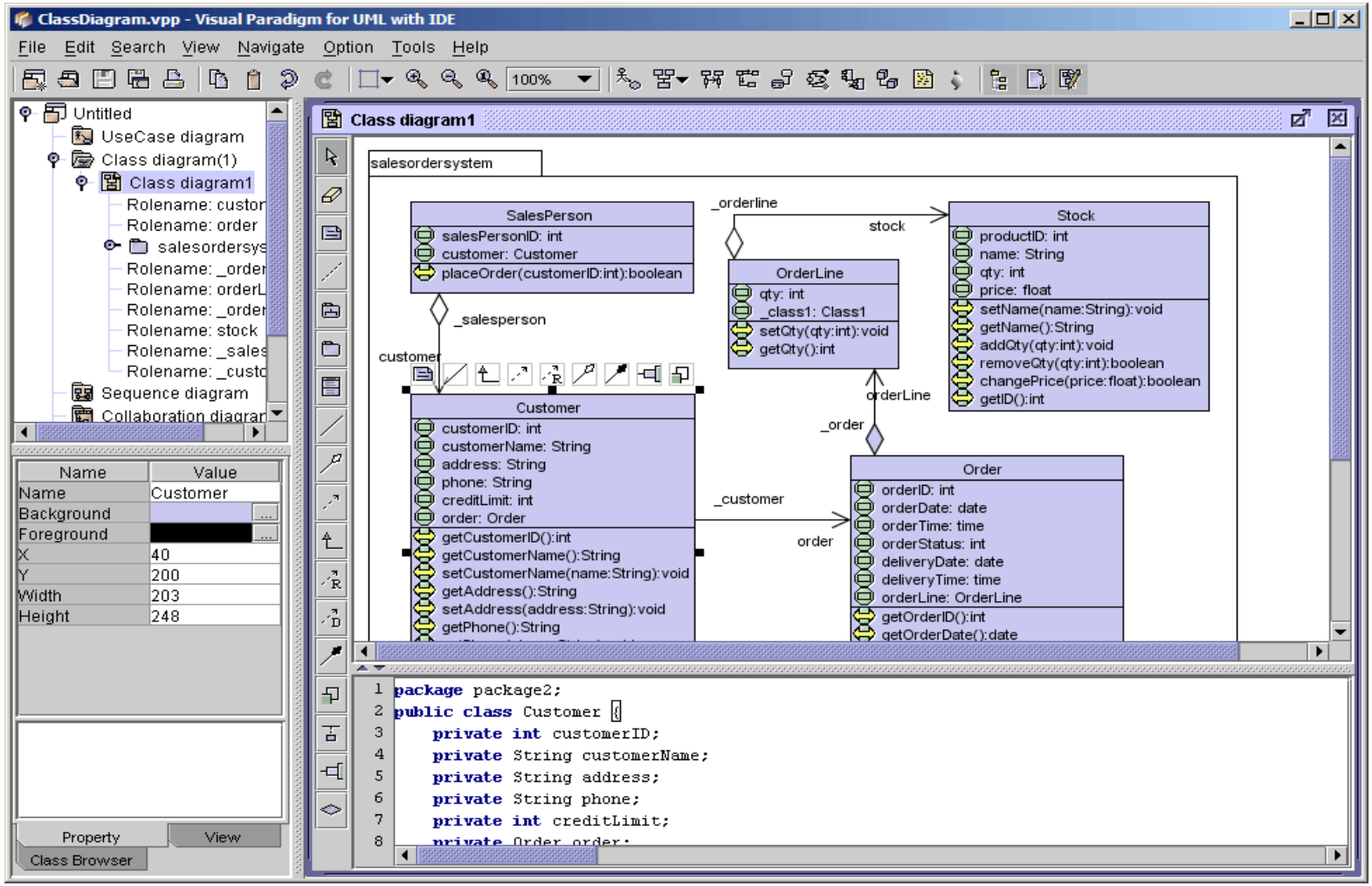


Fig: Sample for visual case tool software

rijalujjwal09@gmail.com

Contd. ….

❖ Types of CASE Tools

- **Diagram Tools:** It helps in diagrammatic and graphical representations of the data and system processes. It represents elements, control flow and data flow among different software components and system structure in a pictorial form. For example: Flowchart Maker tool for making state-of-the-art flowcharts.
- **Computer Display and Report Generators:** It helps in understanding the data requirements and the relationships involved.
- **Analysis Tools:** It focuses on inconsistent, incorrect specifications involved in the diagram and data flow. It helps in collecting requirements; automatically check for any irregularity, imprecision in the diagrams, data redundancies or erroneous omissions.

rijalujjwal09@gmail.com

Contd. ….

- **Central Repository:** It provides the single point of storage for data diagrams, reports, and documents related to the project management.
- **Documentation Generators:** It helps in generating user and technical documentation as per standards. It creates documents for technical users. For example: Doxygen, DrExplain, Adobe RoboHelp for documentation.
- **Code Generators:** It aids in the auto generation of code, including definitions, with the help of the designs, documents and diagrams.

rijalujjwal09@gmail.com

Contd. ….

❖ Advantages of CASE Tools

- As special emphasis is placed on redesign as well as testing, the servicing cost of a product over its expected lifetime is considerably reduced.
- The overall quality of the product is improved as an organized approach is undertaken during the process of development.
- Chances to meet real-world requirements are more likely and easier with a computer-aided software engineering approach.
- CASE indirectly provides an organization with a competitive advantage by helping to ensure the development of the high-quality products, etc.

rijalujjwal09@gmail.com

Contd. ….

❖ Disadvantages of CASE Tools

- **Purchasing CASE tools is not an easy task**. The cost of CASE tools is very high. For this reason, small software development firms do not invest more in CASE tools.
- In general, programmer productivity may fall in the initial phase of implementation as users need time to learn this technology (**Learning Curve**).
- It is important to make proper selection of CASE tools to get maximum benefits from the tools, as the wrong selection may lead to the wrong results indistinctively (**Tool Mix**), etc.

rijalujjwal09@gmail.com

Software Process Lifecycle Models

- A software process model is a simplified description of a software process that presents one view of that process. Process models may include activities that are part of the software process, software products and the roles of people involved in software engineering.
- **Some of the software process lifecycle models are:**

1. **Build and Fix Model**
2. **The Traditional Waterfall Model**
3. **The Prototyping Model**
4. **The Spiral Model**
5. **Rapid Application Development (RAD) Model**
6. **Agile Development Model**

rijalujjwal09@gmail.com

Build and Fix Model

- In this model, first version of the product is built and it is continuously modified till the client is satisfied.
- This is also known as Ad-hoc model. Generally, problems solved with this model are small and is unsatisfactory for product of a reasonable size.
- Product is always in operational mode. It is a single person task and product is constructed without specification. No documentation is done for the software and the maintenance is also ad-hoc.
- This model is not suitable for today's environment where products are developed for people with no computer background.

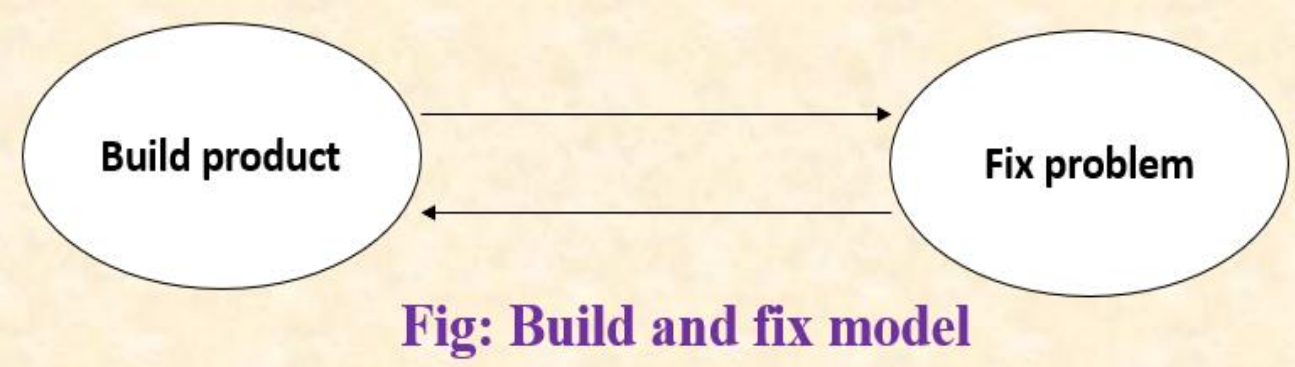


Fig: Build and fix model

rijalujjwal09@gmail.com

Advantages of Build and fix model

- **Model is useful only for small sized projects, in fact only for programming exercise of 100 or 150 lines long.**

Disadvantages of Build and fix model

- **Model is not suitable for larger projects.**
- **Maintenance of software product is extremely difficult in the absence of specification or designed documents.**
- **Higher chances of faults.**
- **Reworking of product results into the increased costs, etc.**

rijalujjwal09@gmail.com

Traditional Waterfall SDLC Model

- **Waterfall model**, sometimes called **Classic life cycle** or **the linear sequential model**, is the oldest and the most widely used paradigm for information systems development. This structured approach looks at the system from a top-down view.
- It is a formalized step by step approach to the SDLC which consists of phases or activities. The activities of one phase must be completed before moving to the next phase.
- At the completion of each activity or phase, a milestone has been reached and a document is produced to be approved by the stakeholders before moving to the next activity or phase; with great care to the amounts of documentation and signoffs through each part of the development cycle is required thoroughly.
- This life cycle model is named as waterfall model because its diagrammatic representation resembles a cascade of waterfall.
- ***The different phases of traditional waterfall model are shown in the below figure:***

rijalujjwal09@gmail.com

Contd. ….

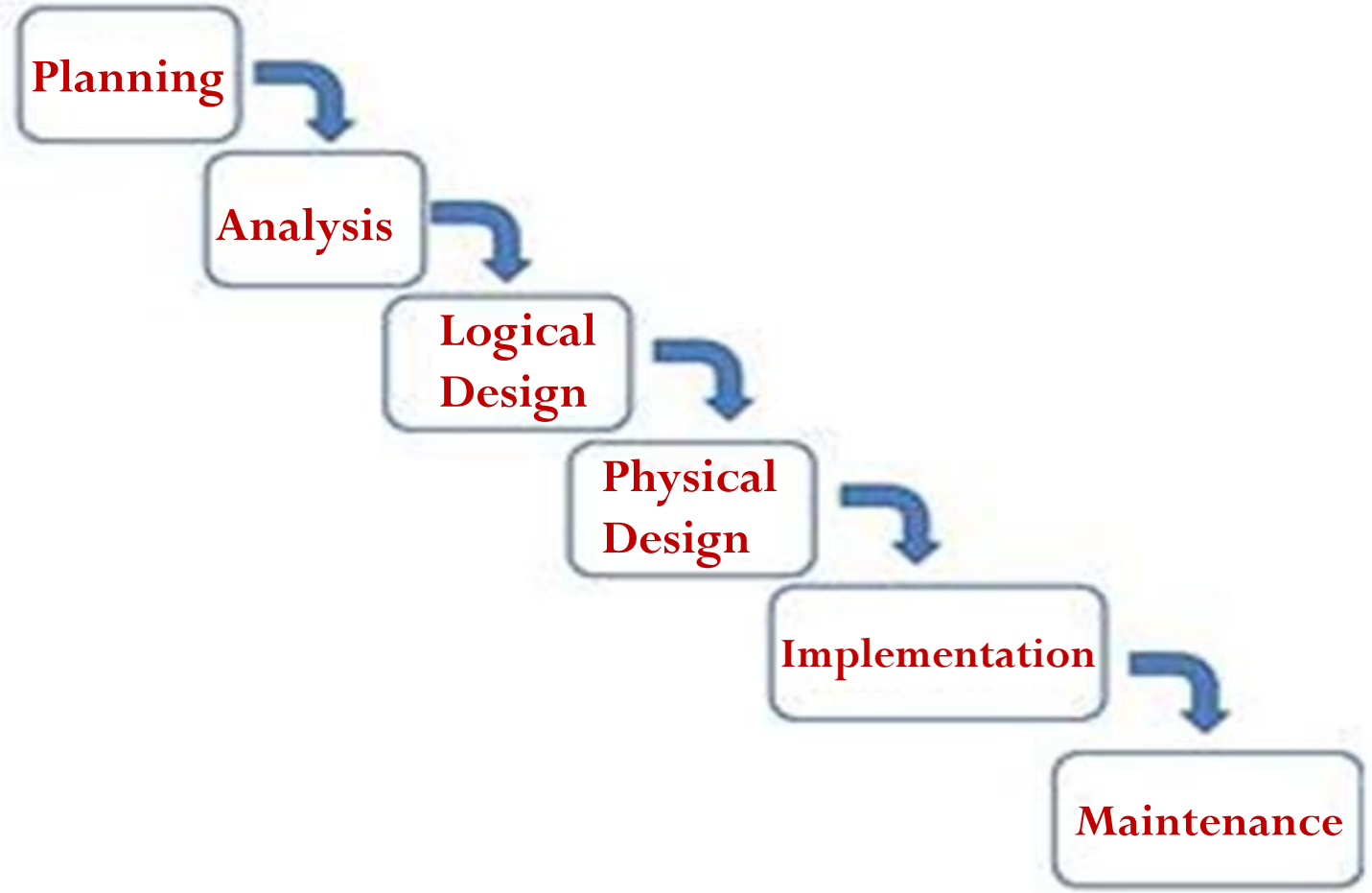


Fig: Traditional Waterfall SDLC Model

rijalujjwal09@gmail.com

Contd. ….

❖ Applications of Waterfall Model

- Every software developed are different and require a suitable SDLC approach to be followed based on the internal and external factors.
- ***Some situations where the use of Waterfall Model is most appropriate are as follows:***

✓ Requirements are very well documented, clear and fixed.

✓ Product definition is stable.

✓ Technology is understood and is not dynamic.

✓ There are no ambiguous requirements.

✓ Ample resources with required expertise are available to support the product.

✓ The project is short, etc.

rijalujjwal09@gmail.com

Contd. ….

❖ Advantages of Waterfall Model

- Simple and easy to understand and use.
- Easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Clearly defined stages and well understood milestones.
- Easy to arrange tasks and, process and results are well documented, etc.

rijalujjwal09@gmail.com

Contd. ….

❖ Disadvantages of Waterfall Model

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model of complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.
- It is difficult to measure progress within the stages and cannot accommodate the changing requirements.
- Adjusting scope during the life cycle can end a project.
- Integration is done as a "big–bang" at the very end, which does not allow identifying the technological or business bottleneck or challenges very early, etc.

rijalujjwal09@gmail.com

❑ Other Approaches

❖ Prototyping Model Approach

- The **prototype model** suggests that before developing the actual software, a working **prototype** of the system should be built. A **prototype** is a partially developed product.
- A **prototype** has limited functional capabilities, low reliability and inefficient performance as compared to the actual software.
- In fact, a **prototype** is a toy and crude implementation of a system. A **prototype** can be built very quickly by using several shortcuts by developing inefficient, inaccurate or dummy functionalities.
- ***The prototyping model has following two types:***

✓ **Throwaway Prototyping:** In this method, developers can explore the ideas as well as get proper customer feedback. Here, the **prototype** need not be a fine one, and so it can further be iterated to develop new versions of the final product.

✓ **Evolutionary Prototyping:** Here, our developed **prototype** will primarily be incremented for refining on the foundation of customer opinion until the final one gets accepted. It provides an improved way which can save time and effort.

rijalujjwal09@gmail.com

Contd. ….

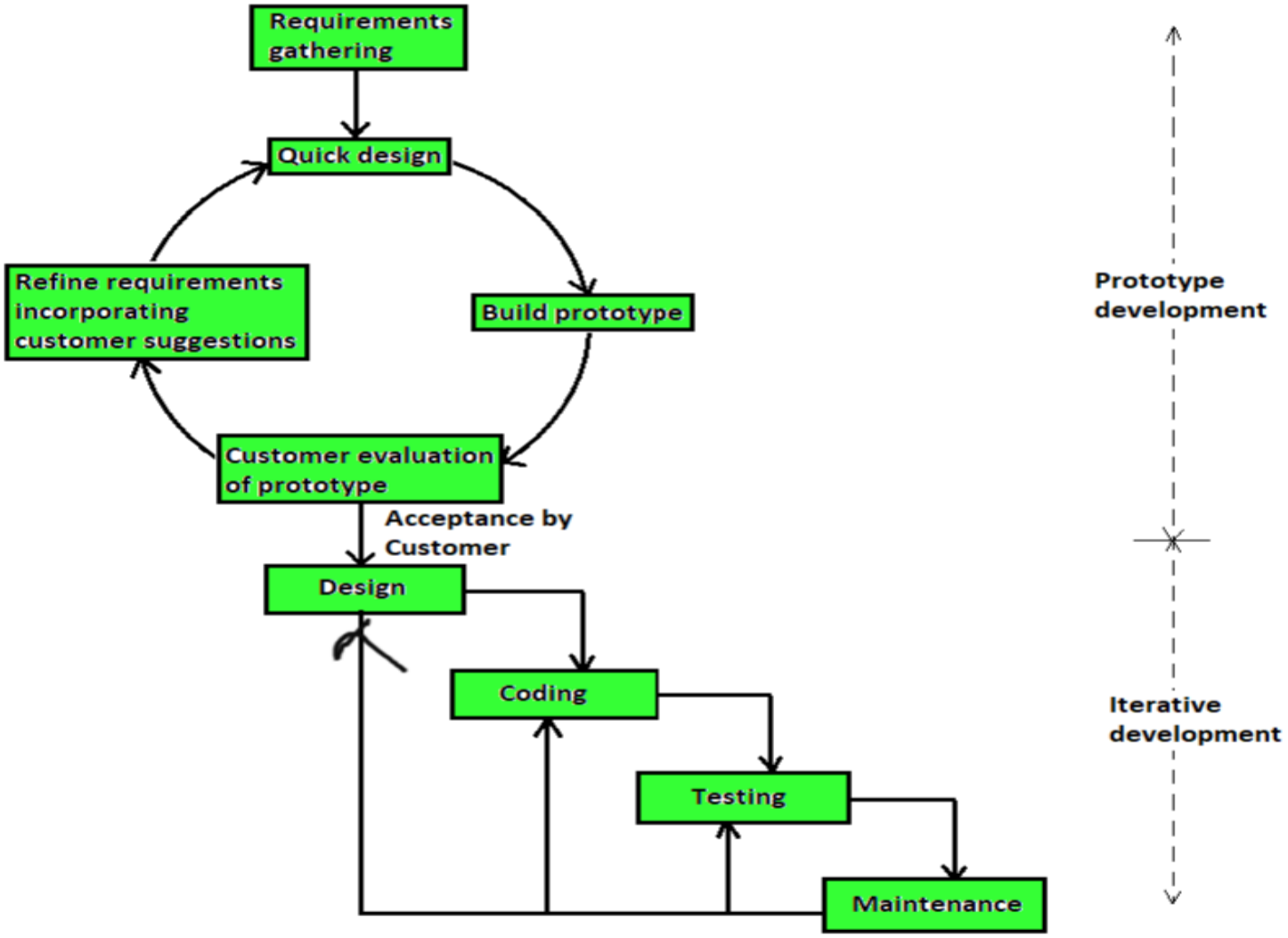


Fig: The Prototyping Model with its various Phases

rijalujjwal09@gmail.com

Contd. ….

- **The software is developed using the prototyping model through two major activities: one is prototype construction and another is iterative waterfall based software development model.**

Prototype development

- It starts with an initial requirements gathering phase. A quick design is carried out and a prototype is built. The developed prototype is submitted to the customer for evaluation.
- Based on the customer feedback, the requirements are refined and the prototype is suitably modified. This cycle of obtaining customer feedback and modifying the prototype continues till the customer approves the prototype.

rijalujjwal09@gmail.com

Contd. ....

Iterative development

- Once the customer approves the prototype, the actual software is developed using the iterative waterfall approach.
- In spite of the availability of a working prototype, the SRS document is usually needed to be developed since the SRS document is invaluable for carrying out tractability analysis, verification and test case during later phases.
- The code for the prototype is usually thrown away. However, the experience gathered from developing the prototype helps a great deal in developing the actual software.
- By constructing the prototype and submitting it for user evaluation, many customer requirements get properly defined and technical issues get resolved by experimenting with the prototype. This minimizes later change requests from the customer and the associated redesign costs.

rijalujjwal09@gmail.com

Contd. ….

Advantages of the Prototyping Model

- ✓ A partial product is built in the initial stage. Therefore, customer gets chance to see the product early in the lifecycle and thus, gives the necessary feedback.
- ✓ It is advantageous to develop the graphical user interface (GUI) part of a software using the Prototyping model.
- ✓ Through prototype, the user can experiment with a working user interface and they can suggest any change if needed.
- ✓ Requirements become more clear.
- ✓ New requirements can be easily accommodated.
- ✓ As user is involved from the starting of the project, he tends to be more comfortable and satisfied, etc.

rijalujjwal09@gmail.com

Contd. ….

Disadvantages of the Prototyping Model

- ✓ Developers in hurry may build prototypes and end up with sub-optimal solutions.
- ✓ After seeing an early prototype, the user may demand the actual system to be delivered sooner.
- ✓ End users may not like to know the differences between the prototyped and well-engineered developed system.
- ✓ If not managed properly, then the interactive process of prototype demonstration and refinement can continue for long duration of time.
- ✓ If end user is satisfied with the initial prototype, then he may lose interest in the project.
- ✓ Prototyping model usually lacks proper documentation.
- ✓ The prototyping model is effective only for those projects for which the risks can be identified before the development of the project starts, etc.

rijalujjwal09@gmail.com

Contd. ….

❖ Spiral Model Approach

- The traditional software lifecycle development model did not deal sufficiently with the uncertainty which is for risk analysis.
- Important software projects have failed because project risks were neglected and nobody was prepared when something unforeseen happen.
- **Dr. Barry Boehm** recognized this and tried to incorporate the project **risk factor into a lifecycle model**. As a result, he proposed a **spiral model** which was presented in **1986**.
- ***The different phases of the spiral model are shown in the below figure:***

rijalujjwal09@gmail.com

Contd. ….

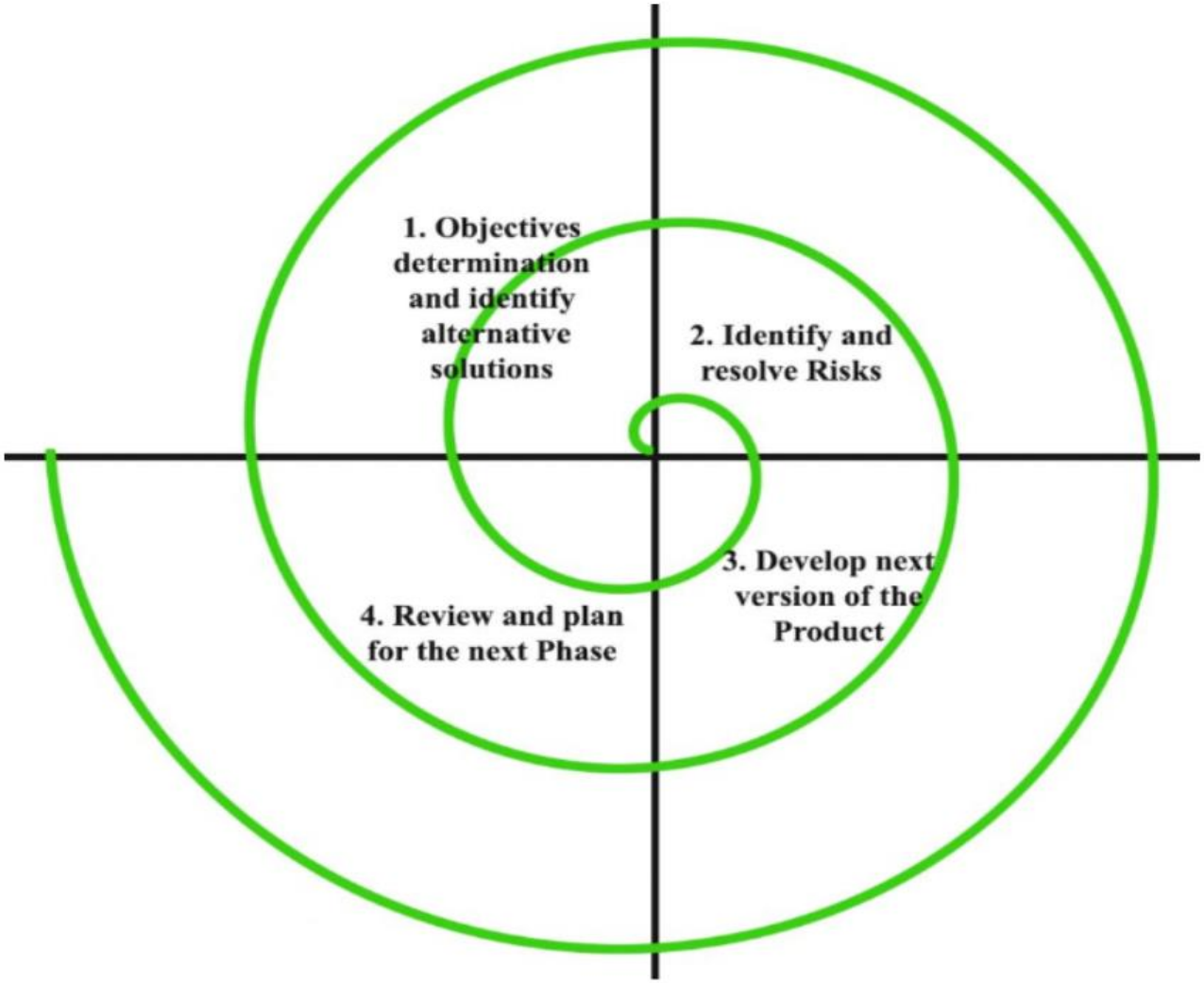


Fig: Spiral Lifecycle Model with its various Phases

rijalujjwal09@gmail.com

Contd. ….

- ***Each phase of spiral model is divided into four quadrants as shown in the above figure.***
- ***The functions of these four quadrants are discussed below:***

I. Objectives determination and identify alternative solutions

- Requirements are gathered from the customers and the objectives are identified, elaborated and analyzed at the start of every phase.
- Then, the alternative solutions possible for the phase are proposed in this quadrant of Spiral model.

II. Identify and resolve risks

- During this second quadrant, all the possible solutions are evaluated to select the best possible solution.
- Then the risks associated with that solution are identified and the risks are resolved using the best possible strategy of risk mitigation and risk management process.
- At the end of this quadrant, prototype is built for the best possible solution.

rijalujjwal09@gmail.com

Contd. ….

III. Develop the next version of the product

- During the third quadrant, the identified features are developed and verified through the testing process. At the end of the third quadrant, the next version of the software is available.
- The typical activities in this quadrant could be creation of design, review of design, development of code, inspection of code, testing, and documenting the project.
- The first build is the customers' first look at the system. ***After this, the planning phase begins.***

IV. Review and plan for the next phase

- In the fourth quadrant, the customers evaluate the so far developed version of the product.
- In the end, planning for the next phase is started. ***That is why, this phase of spiral model is called the planning phase.***
- Typical activities in this quadrant could be development of the project plan, development of the configuration management plan, development of the test plan, and development of the installation plan.

rijalujjwal09@gmail.com

Contd. ….

Advantages of Spiral model

- ✓ Model incorporates software quality objectives into software development which is good for large projects.
- ✓ The risk analysis and validation stage eliminates errors in the early phase of the development.
- ✓ The model makes the use of techniques like reuse, prototyping and component based design.
- ✓ It provides early and frequent feedbacks from the users to developers ensuring a correct product with high quality.
- ✓ Change requests in the requirements at later phase can be incorporated accurately by using this model, etc.

Disadvantages of Spiral model

- ✓ Model is not easily suitable for small projects.
- ✓ Model requires expertise in risk management and excellent management skills.
- ✓ Different persons involved in the project may find it complex to use.
- ✓ The model makes the developers to use components; it requires extra cost,
- ✓ As the number of phases in this model is unknown at the start of the project, so time estimation is difficult, etc.

rijalujjwal09@gmail.com

Contd. ….

Why Spiral Model is called Meta Model?

- The spiral model is called as a meta model because it subsumes all the other SDLC models. For example, a single loop spiral actually represents the iterative waterfall model. The spiral model incorporates the stepwise approach of the classical waterfall model.
- The spiral model uses the approach of prototyping model by building a prototype at the start of each phase as a risk handling technique.
- Also, the spiral model can be considered as supporting for the evolutionary model because the iterations along the spiral can be considered as evolutionary levels through which the complete system is built appropriately.

rijalujjwal09@gmail.com

Contd. ….

❖ Rapid Application Development (RAD) Model Approach

- The RAD model was proposed by IBM in 1980's and later on, was introduced to software community by James Martin through his book. The important feature of RAD model is the increased involvement of users/ customers at all the stages of lifecycle through the use of GUI tools.
- The process starts with building a rapid prototype and is delivered to the customer for use and his feedback. Once the customer validates the rapid prototype after using it, requirement specification is derived and design is done to give the final shape to the product.
- After the product is installed, maintenance of the product is continued by refining the requirements, specification, design or coding phase. RAD model is characterized by the quick turn around time from requirement's definition to the completed system.

rijalujjwal09@gmail.com

Contd. ….

- RAD model follows a sequence of evolutionary system integration or prototypes that are reviewed with the customers, discovering the requirements along the way; the development of each integrated delivery is restricted to a well-defined period of time, usually about 60 days, called a time-box.
- Factors that allow the system to be created in the time period of 60 days without sacrificing quality include the use of high powered development tools, a high reuse factor, and knowledgeable and dedicated resources.
- *The phases of Rapid Application Development (RAD) model are shown in the below figure:*

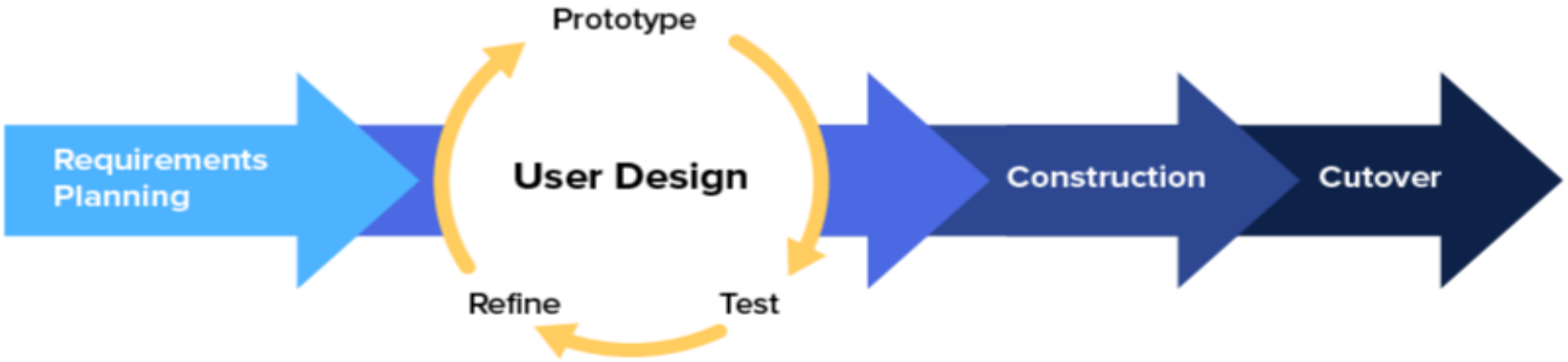


Fig: The RAD Model with its various Phases

rijalujjwal09@gmail.com

Contd. ….

- *The actual working of the phases of the RAD model are discussed below:*

I. Requirements Planning Phase

✓ In this phase, requirements are captured or gathered using any group elicitation techniques.

II. User Design/ User Description Phase

✓ In this phase, joint teams of developers and users are constituted to prepare, understand and review the requirements of the users. The team may use the automated tools and techniques to capture information from the other users/ customers.

III. Construction Phase

✓ This phase combines the detailed design, coding and testing phases of waterfall model. Here, developers release the product of the customers. It is expected to use code generators, screen generators and other types of productivity tools.

IV. Cutover Phase

✓ This phase incorporates the acceptance testing held by the users, installation of the system, screen generators and other types of productivity tools.

rijalujjwal09@gmail.com

Advantages of RAD Model

- A customer is involved at all the stages of development, it leads to a product achieving the customer satisfaction.
- Feedbacks from the customers are available at the initial stages.
- Development time of the product may be reduced due to the use of powerful development tools.
- In order to increase productivity, CASE tools and frameworks are used.
- Fewer developers are needed because there is the use of code generators, CASE tools and frameworks.
- Involvement of the users may increase the acceptability of the products.
- It reuses the existing program components, etc.

rijalujjwal09@gmail.com

Disadvantages of RAD Model

- Highly specialized and skilled developers are required and such developers may not be easily available.
- Model is inefficient and ineffective if the system can not be properly modularized.
- If users cannot be involved throughout the development, this may not be an appropriate model.
- Absence of reusable components can lead to the failure of the products.
- It requires developers and customers who are committed to rapid fire activities in an abbreviated time frame, etc.

rijalujjwal09@gmail.com

Contd. ....

❖ Agile Development Model Approach

- The meaning of **Agile** is **Swift** or **Versatile**. Agile process model refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.
- The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.
- Each iteration is considered as a short time **frame** in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.
- Each iteration is intended to be small and easily manageable and that can be completed within a couple of weeks only. At a time one iteration is planned, developed and deployed to the customers. Long-term plans are not made. Agile model is the combination of iterative and incremental process models.

rijalujjwal09@gmail.com

Contd. ….

➢ Basic Principles of Agile Model

- To establish close contact with the customer during development and to gain a clear understanding of various requirements, each Agile projects usually includes a customer representative on the team.
- Agile model relies on working software deployment rather than comprehensive documentation.
- Frequent delivery of incremental versions of the software to the customer representative in intervals of few weeks.
- Requirement change requests from the customer are encouraged and efficiently incorporated.
- It emphasizes on having efficient team members and enhancing communications among them is given more importance.
- It is recommended that the development team size should be kept small (5 to 9 people) to help the team members meaningfully engage in face-to-face communication and have collaborative work environment.
- Agile development process usually deploy **Pair programming**. **In pair programming**, two programmers work together at one work-station. One does coding while the other reviews the code as it is typed in. The two programmers switch their roles every hour or so, etc.

rijalujjwal09@gmail.com

Contd. ….

Fig: Agile Development Model

rijalujjwal09@gmail.com

Contd. ….

- *Following are the phases of the Agile model:*

i. Requirements gathering

In this phase, we must define the requirements. We should explain the business opportunities and plan the time and effort needed to build the project. Based on this information, we can evaluate technical and economic feasibility.

ii. Design the requirements

When we have identified the project, work with stakeholders to define the requirements. We can use the user flow diagram or the high level UML diagram to show the work of new features and show how it will apply to our existing system.

iii. Construction/ Iteration

When the team defines the requirements, the work begins. Designers and developers start working on their project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionalities.

iv. Testing

In this phase, the Quality Assurance (QA) team examines the product's performance and looks for the bug.

v. Deployment

In this phase, the team issues a product for the user's work environment.

vi. Feedback

After releasing the product, the last step is feedback. In this phase, the team receives feedback about the product and works through the feedback.

rijalujjwal09@gmail.com

Contd. ….

A few Agile SDLC models are as follows:

- Extreme Programming (XP)
- Scrum
- Crystal
- Feature Driven Development (FDD)
- Dynamic Software Development Method (DSDM)
- Lean software Development
- Unified Process, etc.

When to use the Agile Model?

- When frequent changes are required.
- When a highly qualified and experienced team is available.
- When a customer is ready to have a meeting with a software team all the time.
- When the project size is small, etc.

rijalujjwal09@gmail.com

Contd. ….

Advantages/ Merits/ Pros of Agile Model

- Frequent delivery.
- Face-to-Face communication with the clients.
- Efficient design and fulfills the business requirements.
- Anytime changes are acceptable.
- It reduces the total development time.
- Working through pair programming produces well written compact programs which has fewer errors as compared to programmers working alone, etc.

Disadvantages/ Demerits/ Cons of Agile Model

- Due to the shortage of formal documents, it creates confusion and crucial decisions taken throughout various phases can be misinterpreted at any time by different team members.
- Due to the lack of proper documentation, once the project completes and the developers are allotted to another project, maintenance of the finished project can become a problem, etc.

rijalujjwal09@gmail.com

Introduction to Object–Oriented Development

- **Object–Oriented Analysis and Design (OOAD)** is often called the third approach to the systems development, after the process–oriented and data–oriented approaches.
- The OOAD approach combines data and processes (called methods) into single entities called objects. Objects usually correspond to the real things an information system deals with, such as customers, suppliers, contracts, rental agreements, and other prospects.
- The goal of OOAD is to make systems' elements more reusable, thus improving system quality and the productivity of systems analysis and design.
- In other words, the term object oriented describes the system as a collection of discrete objects that incorporate both data structure and behaviour.
- It is a way of thinking about problems using models organized around real world concepts.
- OOAD promotes the better understanding of requirements, clean design and more maintainable system.

rijalujjwal09@gmail.com

❑ Basic Characteristics of Object–Oriented System

- **Class:** Objects with same data structure and behaviour are grouped into a class. A class is an abstraction that describes properties important to an application and ignores the rest i.e. a class is a collection of similar and related objects.
- **Abstraction:** Classes are built on the basis of abstraction, where a set of similar objects are observed and their common characteristics are listed. Of all these, the characteristics of concern to the system under observation are picked up and the class definition is made. The attributes of no concern to the system are left out. This is know as abstraction. The abstraction of an object varies according to its application.
- **Polymorphism:** In Greek, **poly** means **many** and **morphism** means **forms**. Polymorphism means the same operation may behave differently for different classes.

rijalujjwal09@gmail.com

Contd. ….

- **Inheritance:** Inheritance is the sharing of attributes and operations among classes based on hierarchical relationship. A supper class has general information that sub – classes refine and elaborate. Each sub – class incorporates or inherits all the features of its super class and adds its own unique features. This concept is used to apply the idea of **code reusability** of the objects.
- **Reusability:** The classes once defined can easily be used by other applications. This is achieved by defining classes and putting them into a library of classes where all the classes are maintained for future use. Whenever a new class is needed, the programmer looks into the library of classes and if it is available, it can be picked up directly from there.
- **Encapsulation and data hiding:** Encapsulation is a technique that allows the programmer to hide the internal functioning of the objects from the users of the objects. Encapsulation separates the internal functioning of the object from the external functioning, thus providing the user flexibility to change the external behaviour of the object making the programmer code safe against the changes made by the user.

rijalujjwal09@gmail.com

❑ Object–Oriented Development Life Cycle

1. System Analysis

- As in any other system development model, system analysis is the first phase of development in case of object modelling too.
- In phis phase, the developer interacts with the user of the system to find out the user requirements and analyses the system to understand the functioning.
- ***This methodology supports and uses three basic models:***

I. Object Model: This model describes the objects in a system and their interrelationships. This model observes all the objects as static and does not pay attention to their dynamic nature.

II. Dynamic Model: This model depicts the dynamic aspects of the system. It portrays the changes occurring in the states of various objects with the events that might occur in the system.

III. Functional Model: This model describes the data transformations of the system. This describes the flow of data and the changes that occur to the data throughout the system.

rijalujjwal09@gmail.com

Contd. ….

2. System Design

- System design is the next development stage where the overall architecture of the desired system is decided. The system is organized as a set of sub-systems interacting with each other.
- While designing the system as a set of interacting sub-systems, the analyst takes care of specifications as observed in system analysis as well as what is required out of the new system by the end user.
- While designing the system, the stress lies on the objects comprising the system and not on the processes being carried out in the system as in the case of traditional Waterfall model, where the processes form the important part of the system.

rijalujjwal09@gmail.com

Contd. ....

3. Object Design

- In this phase, the details of the system analysis and system design are implemented. The objects identified in the system design phase are designed.
- Here, the implementation of these objects is decided as the data structures get defined and also the interrelationships between the objects are defined.
- Objected oriented philosophy is very much similar to real world and hence is gaining popularity as the systems here are seen as a set of interacting objects as in the real world.
- Just as every programming languages provide various data types and various variables of that type can be created, similarly, in case of objects, certain data types are predefined.

4. Implementation

- During this phase, the class objects and the interrelationships of these classes are translated and actually coded using the programming language decided upon.
- The databases are made and the complete system is given a functional shape.

rijalujjwal09@gmail.com

❑ Object Oriented Design using the UML

- A design model in software engineering is an object-oriented approach that represents the system.
- It shows the object or object-class relationship in the system.
- In fact, design models show the association or relationship between the system entities that act as the bridge between the requirement specification and implementation of the system.
- The design model that we choose to design a system depends on the type of system being developed.
- **In object-oriented design, we use UML diagrams to represent the system.**
- **Unified Modeling Language (UML)** is a general purpose modelling language whose aim is to define a standard way to visualize the way a system has been designed.
- **UML is not a programming language, it is rather a visual language**. We use UML diagrams to portray the behaviour and structure of the system.
- **The Object Management Group (OMG) adopted Unified Modeling Language (UML) as a standard in 1997.**
- **Later on, International Organization for Standardization (ISO) published UML as an approved standard in 2005.**

rijalujjwal09@gmail.com

Types of UML Diagrams

- In UML, a range of different models may be produced. The UML provides various models or diagrams that may be produced to document the design.
- **Mainly, there are two kinds of models:**

1. Static Model / Structural Diagrams.

2. Dynamic Model / Behavioral Diagrams.

rijalujjwal09@gmail.com

Types of UML Diagrams

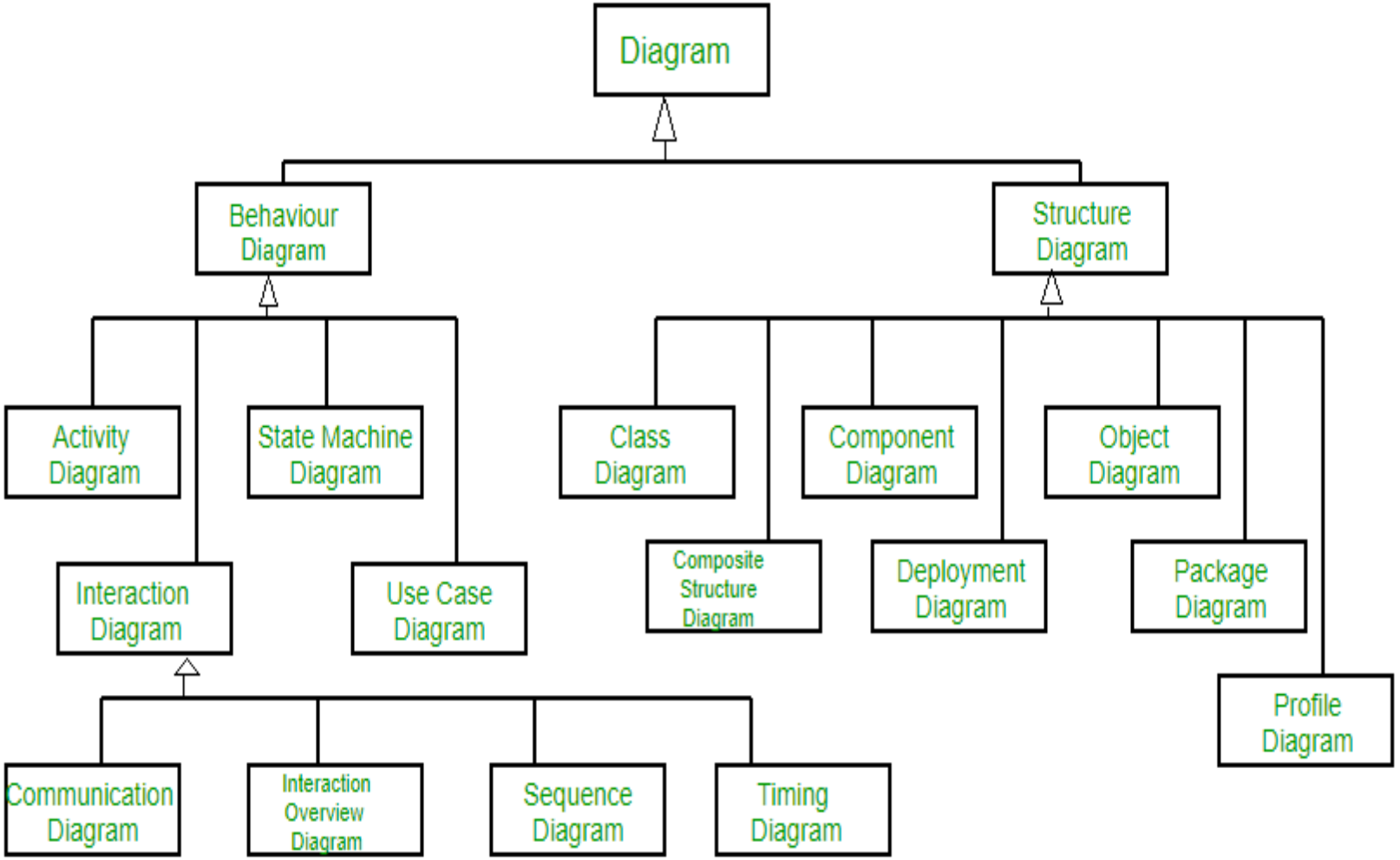


Fig: UML Diagrams

rijalujjwal09@gmail.com

Contd. …

1. Static Model / Structural Diagrams

- It captures the static aspects or structure of a system i.e. shows how the system is structured, including the classes, objects, packages, components, etc. in the system and the relationships between those elements.
- These static aspects represent those parts of a diagram, which forms the main structure and are therefore stable.
- **Structural diagrams include: class diagrams, component diagrams, object diagrams, deployment diagrams, etc.**

I. Class Diagram

- Class diagram is a static diagram which represents the static view of an application.
- Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing the executable code of software application.
- The class diagram describes the attributes and methods (operations) of a class and also the constraints imposed on the system.
- **The purpose of class diagram includes:** analysis and design of the static view of application, describing responsibilities of a system, base for the component and deployment diagrams, forward and reverse engineering, etc.

rijalujjwal09@gmail.com

Contd. …

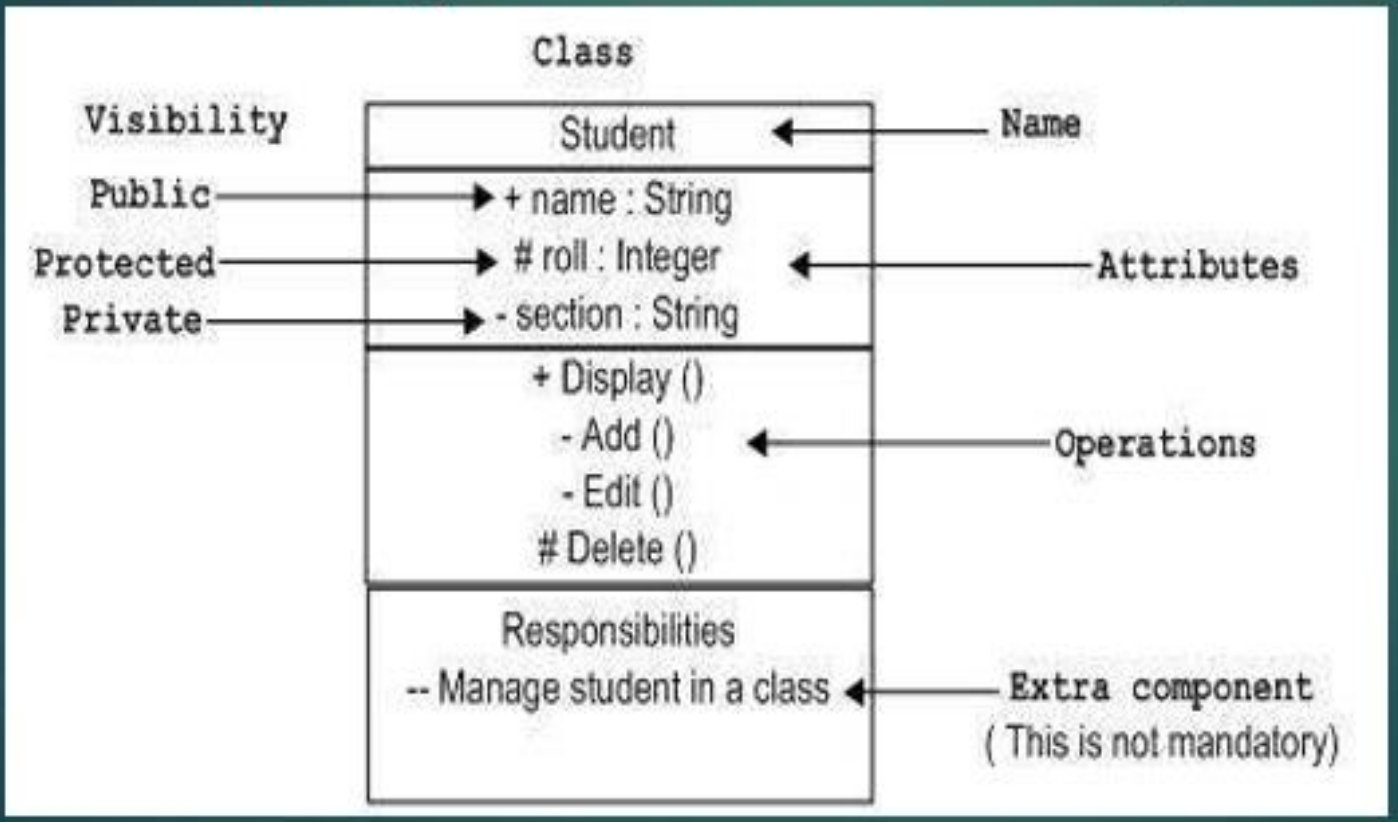
✓ **Symbols**

A. Class Symbol

B. Visibility – a. **Public(+)**: accessed from outside the class.

b. **Private(-)**: only be accessed from within the class.

c. **Protected(#)**: accessed from within the class or any descendant.



rijalujjwal09@gmail.com

Contd. …

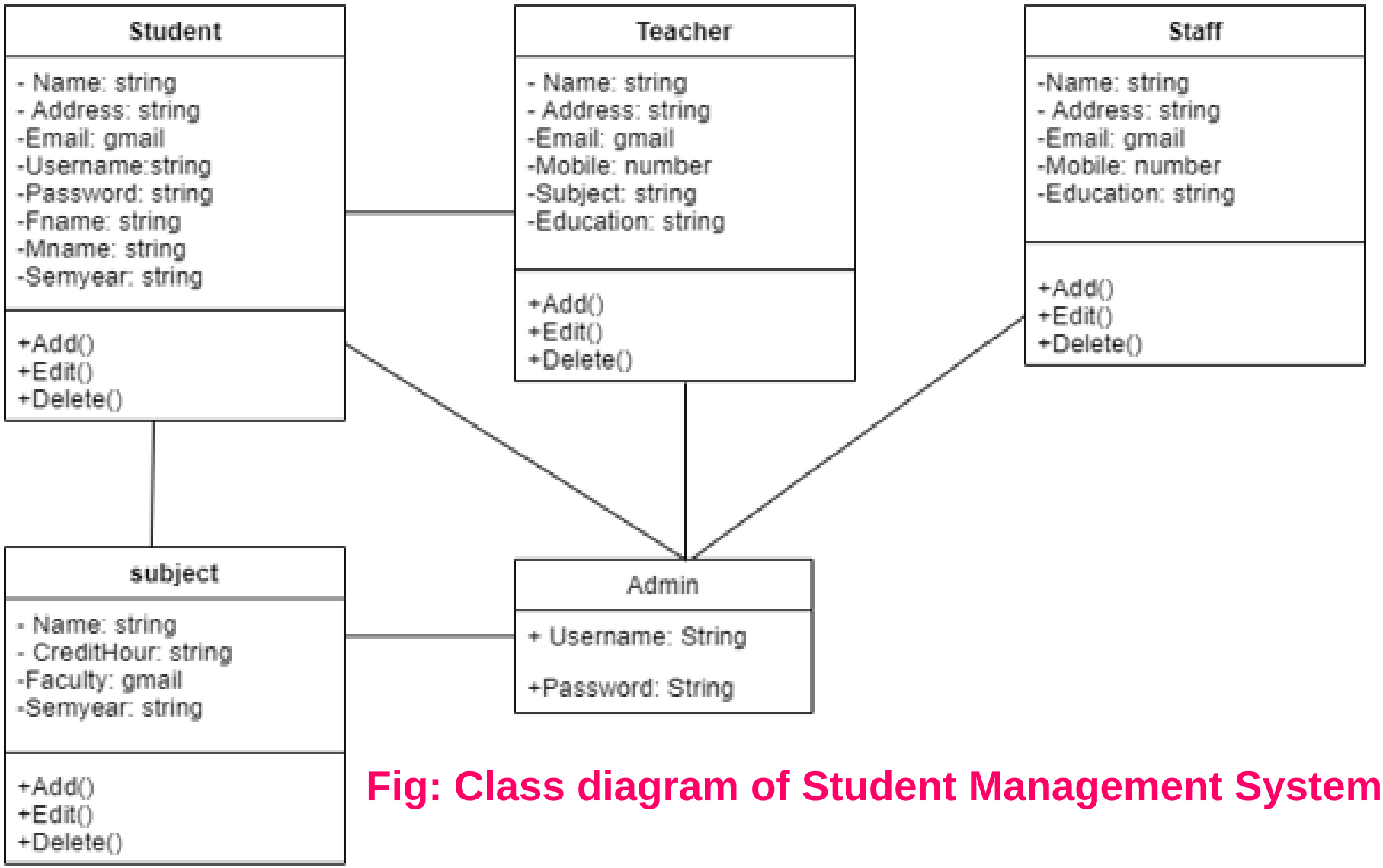


Fig: Class diagram of Student Management System

rijalujjwal09@gmail.com

Contd. …

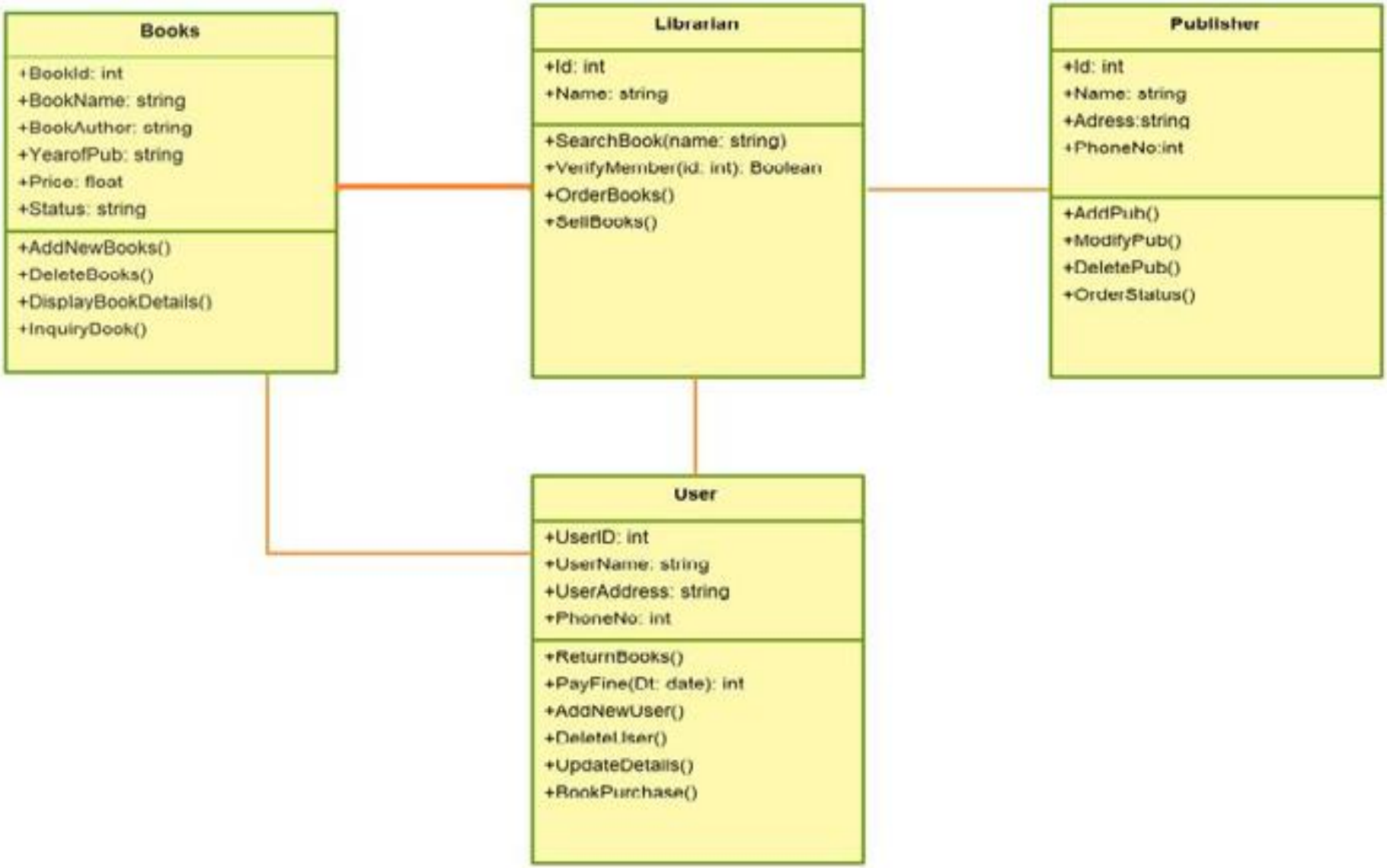


Fig: Class diagram of Library Management System

rijalujjwal09@gmail.com

Contd. …

2. Dynamic Model / Behavioral Diagrams

- It captures the dynamic aspects or behaviour of the system i.e. shows the dynamic structure of the system which forecasts the interactions between the system objects.
- Dynamic aspect can be described as the changing / moving parts of a system.
- Behavioral diagrams assist in understanding and communicating how elements interact and collaborate to provide the functionality of a system.
- **Behavioral diagrams include: Use-Case diagrams, state diagrams, activity diagrams, sequence diagrams, etc.**

I. Use-Case Diagram

- Use-case diagrams are used to depict the functionality of a system or a part of the system.
- They are widely used to illustrate the functional requirements of the system and its interaction with the external agents i.e. actors.
- A use case is a set of scenarios that describes the interaction between a user and a system.
- The two main components of use case diagrams are actors and use cases.
- So a use case diagram displays the relationship among the actors and the use cases.

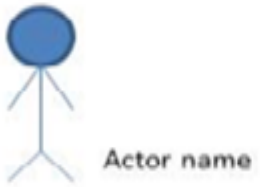
rijalujjwal09@gmail.com

Contd. …

- A use case diagram gives us a high level view of what the system or a part of the system does without going into the implementation details.
- **The purpose of use case diagram includes:**
  - ✓ Used to get the requirement of a system.
  - ✓ Used to get an outside view of a system.
  - ✓ Identify the external and internal factors influencing the system.
  - ✓ Show the interaction among the requirements and the actors.
  - ✓ Focus on the functional requirements, etc.

rijalujjwal09@gmail.com

Contd. ...

1. 

Actor — It is an external entity that interacts with the system. It is represented as the stick sized entity. It can be human or machine. For example: e-Sewa, PayPal, sms server, etc.

2. 

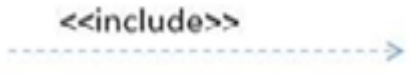
Use case — A use case is an action that the user performs within the system or the system performs on the user.

3. 

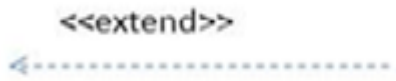
Boundary — System boundary is used to represent the system scope.

4.

Straight Line — Straight line is used to show communication or interaction between the actor and the system.

5. 

Include relationship between use cases

6. 

Extend relationship between use cases

Fig: Symbols used in Use-Case Diagram

rijalujjwal09@gmail.com

Contd. …

Fig: Use-Case Diagram of House Price Prediction

rijalujjwal09@gmail.com

Contd. …

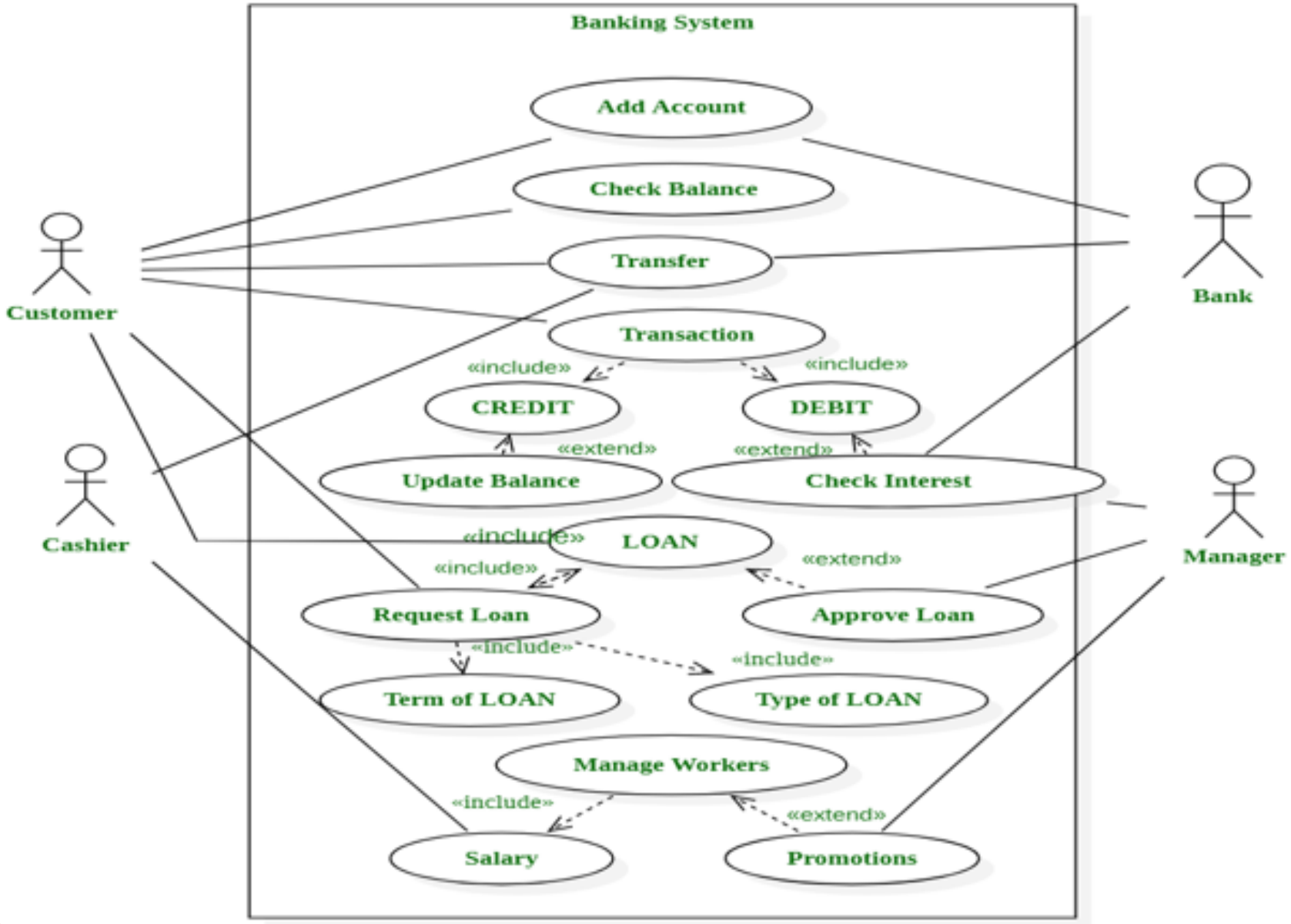


Fig: Use-Case Diagram of Banking System

rijalujjwal09@gmail.com

Contd. …

II. Activity Diagram

- Activity diagrams are basically used for business process modelling, and for modelling the logic captured by a single use case or usage scenario or for modelling the detailed logic of a business rule.
- Activity diagram is basically a flow chart to represent the flow from one activity to another activity.
- The activity can be descried as an operation of the system. So the control flow is drawn from one activity to another, this flow can be sequential or concurrent.
- **The purpose of activity diagrams include:**

✓ Draw the activity flow of a system.
✓ Describe the sequence from one activity to another.
✓ Describe the parallel, branched and concurrent flow of a system, etc.

rijalujjwal09@gmail.com

Contd. …

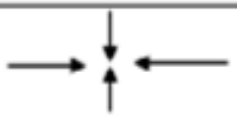
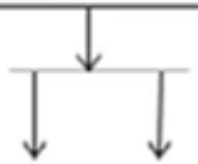
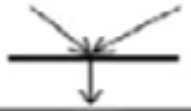
Sr. No	Name	Symbol
1.	Start node	
2.	Action State	
3.	Control Flow / Direction	
4.	Decision node / Choice	
5.	Fork	
6.	Join	
7.	End node	

Fig: Symbols used in Activity Diagram

rijalujjwal09@gmail.com

Contd. …

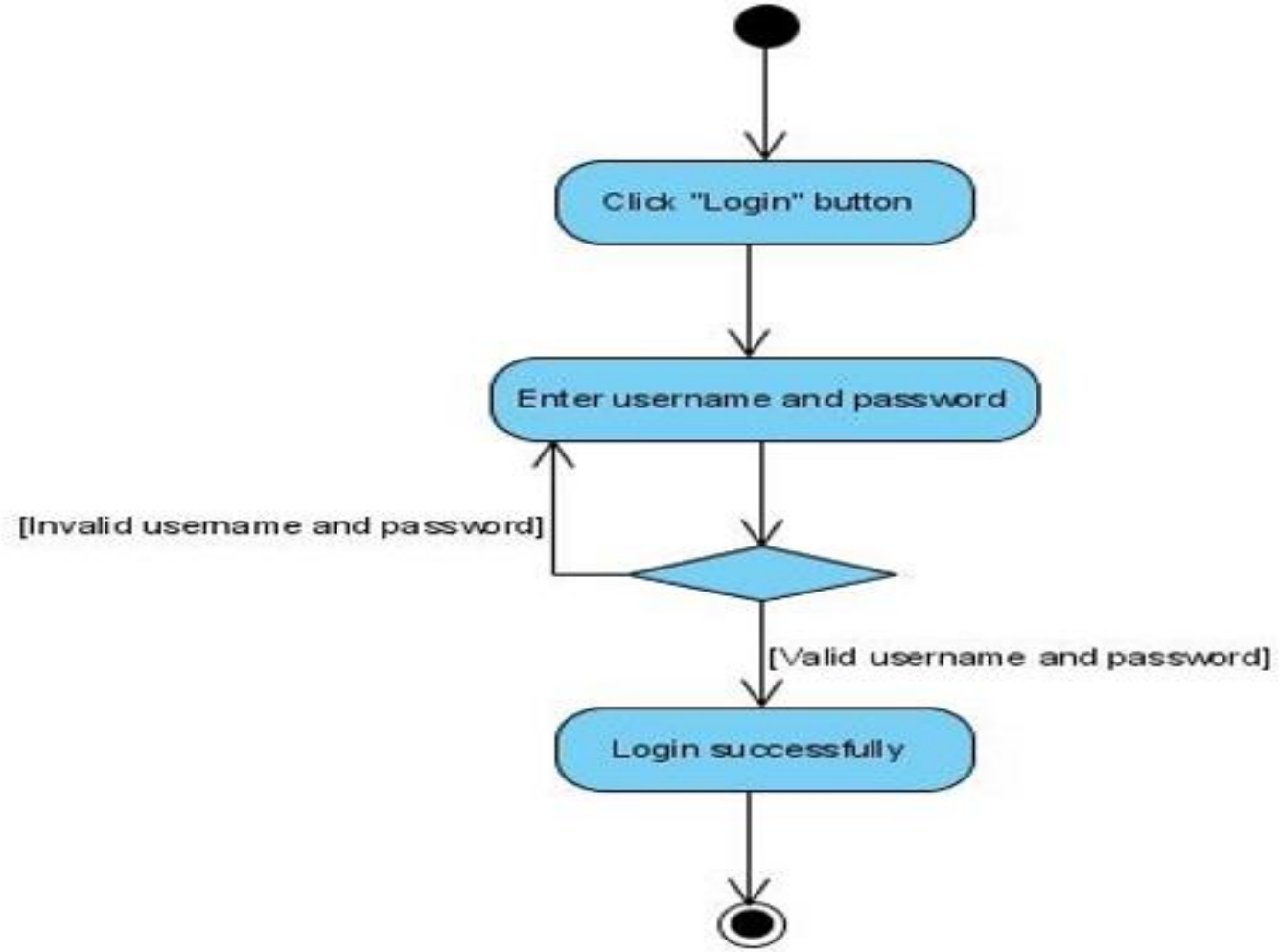


Fig: Activity Diagram of Login Panel

rijalujjwal09@gmail.com

Contd. …

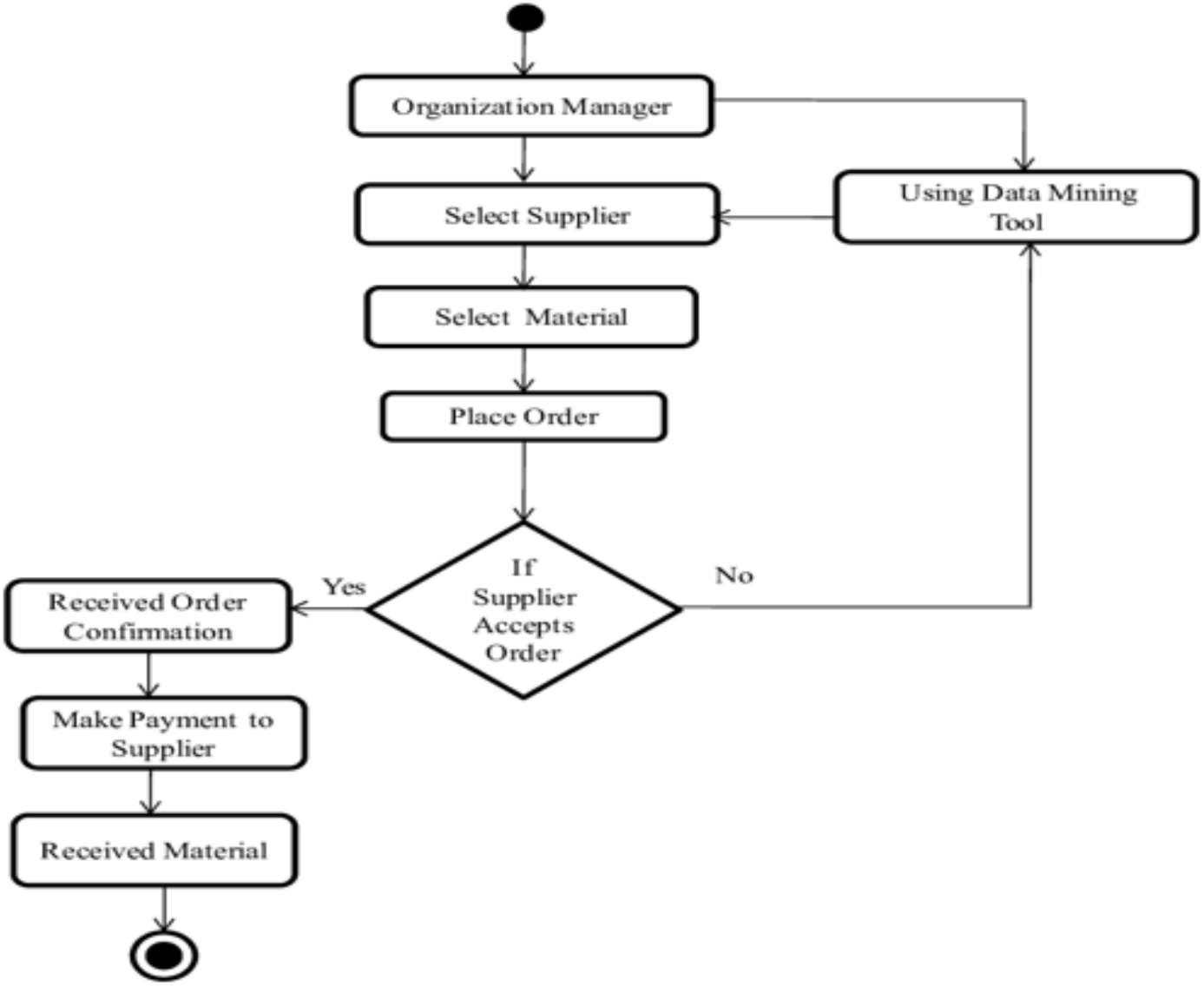


Fig: Activity Diagram of e-Procurement System

rijalujjwal09@gmail.com

Contd. …

III. Sequence Diagram

- A sequence diagram is a form of interaction diagram which shows objects as lifelines running down the page with their interaction over time, represented message drawn as arrows from the source lifelines to the target lifeline.
- Sequence diagrams are good at showing which objects communicate with which other objects and what messages trigger those communications.
- Sequence diagrams are not intended for showing complex procedural logic.
- A repetition or loop within a sequence diagram is depicted as a rectangle, and place the condition for exiting the loop inside the rectangle in brackets.
- **The purpose of sequence diagrams includes:**

✓ Used primarily to show the interactions between the objects in the sequential orders.
✓ Transition from requirements expressed to the neat level of refinement.
✓ Used to document how object in existing software currently interacts, etc.

rijalujjwal09@gmail.com

Contd. …

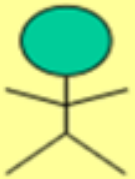
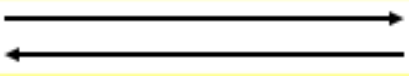
AN ACTOR	
AN OBJECT	anObject:aClass
A LIFELINE	
A FOCUS OF CONTROL	
MESSAGE PASSING (Asynchronous / Synchronous)	
OBJECT DESTRUCTION	x
MESSAGE RETURN	

Fig: Symbols used in Sequence Diagram

rijalujjwal09@gmail.com

Contd. ...

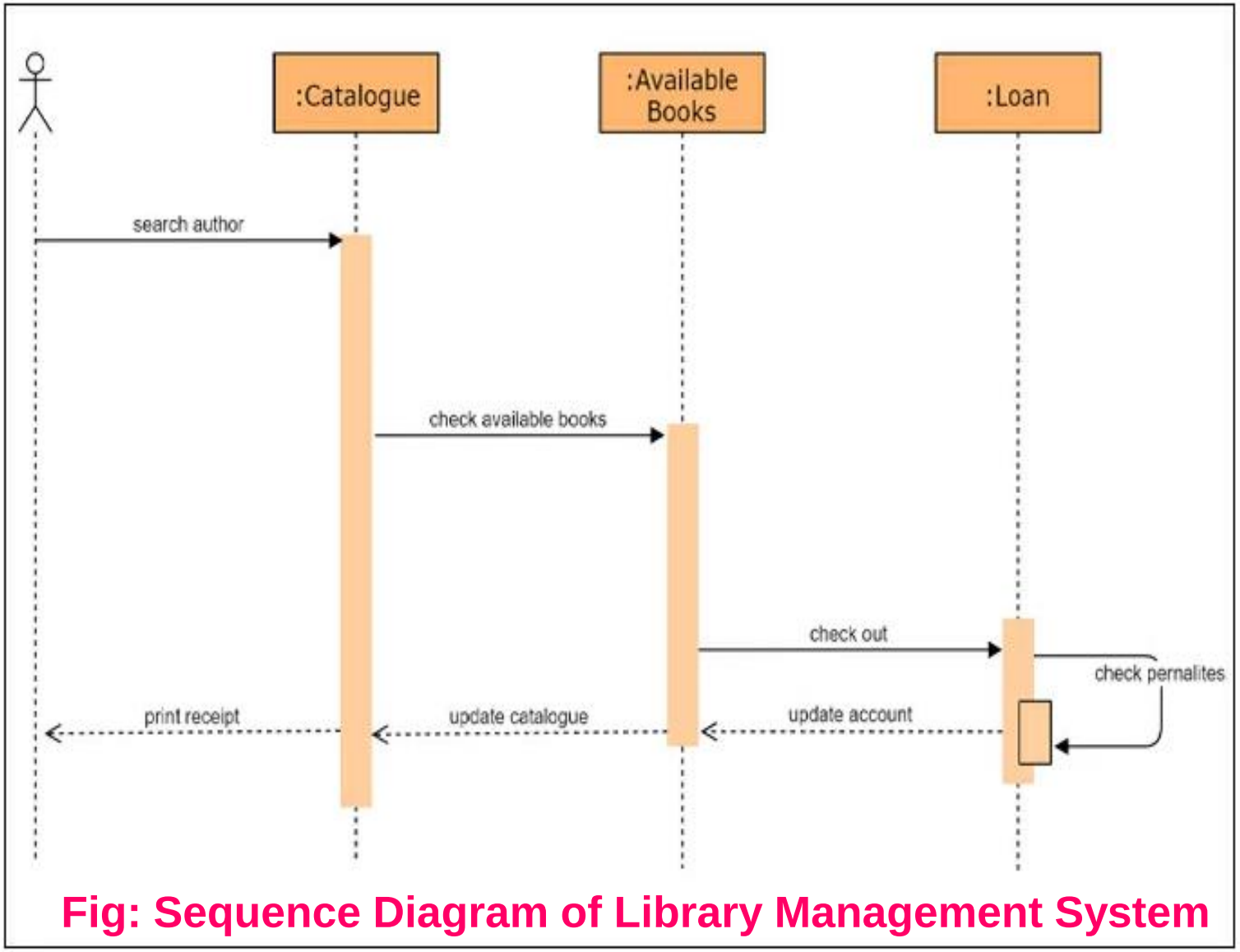


Fig: Sequence Diagram of Library Management System

rijalujjwal09@gmail.com

Contd. ...

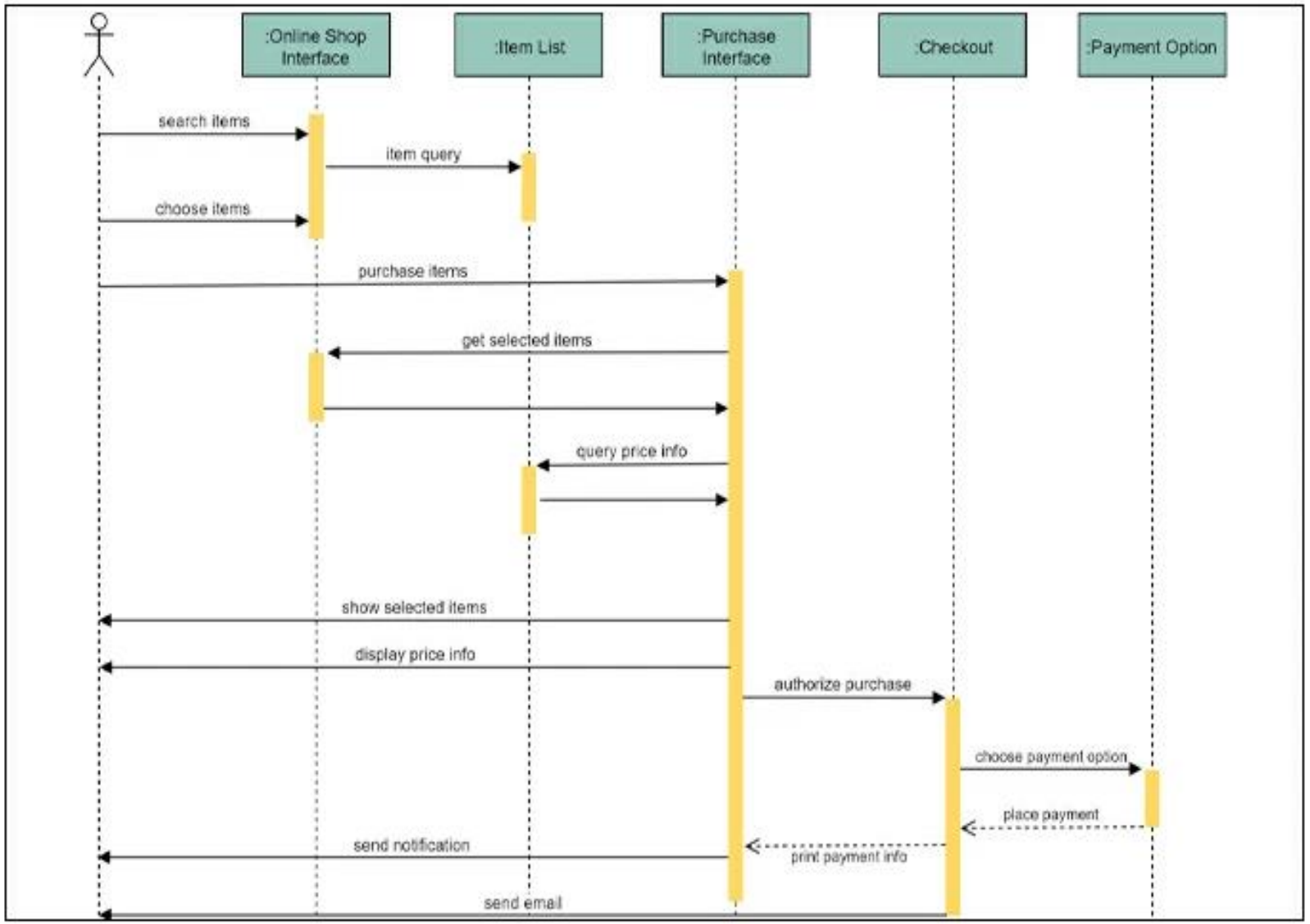
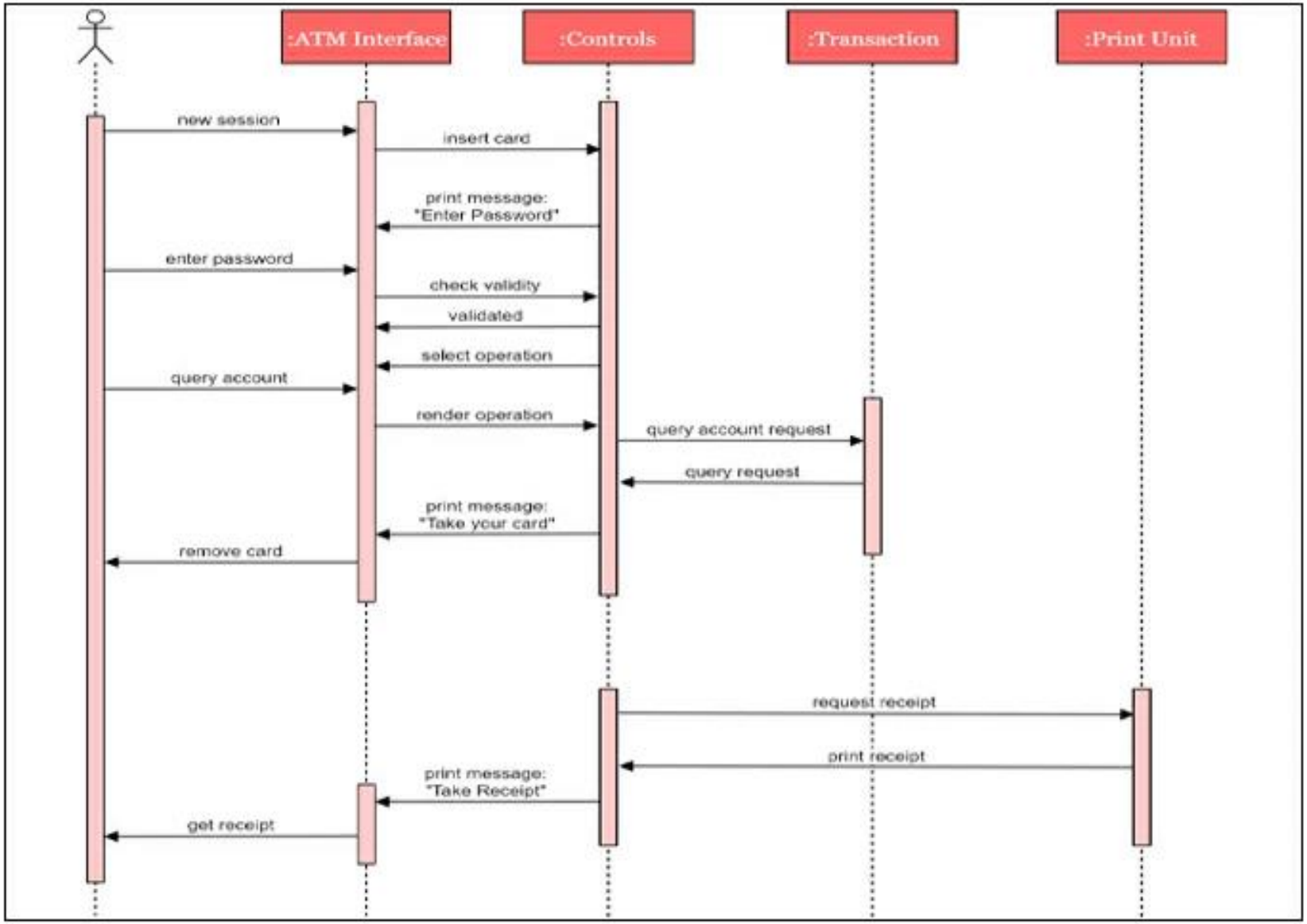


Fig: Sequence Diagram of Online Shopping System

rijalujjwal09@gmail.com

Contd. ...



rijalujjwal09@gmail.com

Fig: Sequence Diagram of ATM System

❑ Managing the Information Systems Project

- **Project management** is an important aspect of development of information system and a skill for a system analyst.
- The focus of the project management is to ensure that system development projects meet the customers' expectations and are delivered within budget and time constraints.
- **A project** is a planned undertaking of related activities to reach the objectives that has a beginning and an end.
- **A project manager (PM)** is a system analyst with a diverse set of skills, such as management, leadership, conflict management, technical management and customer relationship management.
- A project manager is responsible for initiating, planning, executing, monitoring, controlling and closing down a project.
- ***A deliverable is an end product of the SDLC phases.***

rijalujjwal09@gmail.com

❑ Project Triangle

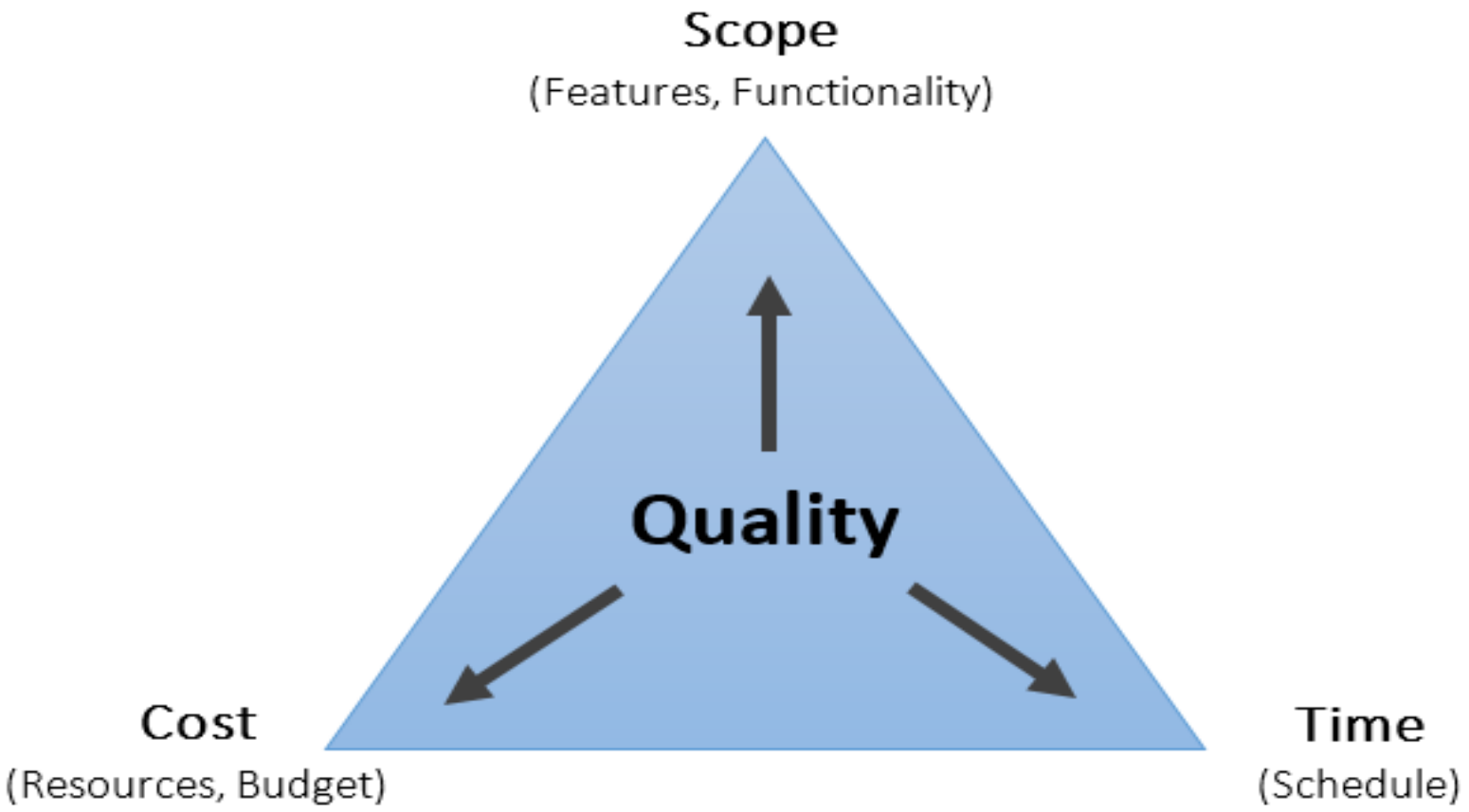


Fig: A typical project triangle includes cost, scope, time, and quality.

rijalujjwal09@gmail.com

Common Activities of a Project Manager

Activity	Description
Leadership	Influencing the activities of others toward the attainment of a common goal through the use of intelligence, personality, and abilities
Management	Getting projects completed through the effective utilization of resources
Customer relations	Working closely with customers to ensure that project deliverables meet expectations
Technical problem solving	Designing and sequencing activities to attain project goals
Conflict management	Managing conflict within a project team to assure that conflict is not too high
Team management	Managing the project team for effective team performance
Risk and change management	Identifying, assessing, and managing the risks and day-to-day changes that occur during a project

rijalujjwal09@gmail.com

❑ Phases/ Activities of Project Management

- Project management is a controlled process of initiating, planning, executing, monitoring and controlling, and closing down a project.
- **Five phases of project management process or Project life cycle phases are discussed below:**

rijalujjwal09@gmail.com

Contd. ….

I. Initiating

- During this phase, the project is conceptualized and feasibility is determined.
- Some activities that should be performed during this process include defining the project goal, defining the project scope, identifying the project manager and the key stakeholders, identifying potential risks, and producing an estimated budget and timelines.

II. Planning

- Next, the project manager will create a blueprint to guide the entire project form ideation through completion.
- This blueprint will map out the project's scope, resources required to create the deliverables, estimated time and financial commitments, communication strategy to ensure stakeholders are kept up to date and involved, execution plan, and proposal for ongoing maintenance.
- If the project has not yet been approved, this blueprint will serve as a critical path of the project life cycle.

rijalujjwal09@gmail.com

Contd. ….

III. Executing

- During this phase, the project manager will conduct the procurement required for the project as well as the staff teams.
- Further, execution of the project objectives requires effective management of the team members.
- The project managers are responsible for delegating and overseeing the work on the project while maintaining good relationships will all the team members and keeping the entire project on time and budget.
- The PM must therefore be highly organized and an exceptional leader.
- That is why because they will need to address team concerns and any issues that arise along the way, requiring frequent and open communication with all the team members and stakeholders.

rijalujjwal09@gmail.com

Contd. ….

IV. Monitoring and Control

- During this phase, the project managers will closely measure the progress of the project to ensure in order to ensure that it has been developing properly.
- Documentation such as data collection, and verbal and written status reports may be used for this phase. Monitoring and controlling is closely related to project planning.
- While planning determines what is to be done, whereas, monitoring and controlling establish how well it has been done.
- Monitoring will detect any necessary corrective actions or changes in the project to keep the project on track.

rijalujjwal09@gmail.com

Contd. ….

V. Closing

- The closing process occurs once the project deliverables have been produced and the stakeholders validate and approve them.
- During this phase, the project manager will close the contracts with suppliers, external vendors, consultants, and other third – party providers.
- All documentations will be archived and a final project report will be produced.
- Further, the final part of the project plan – the plan for troubleshooting and maintenance is conducted after the delivery of the product, produced through the conduction of different phases of project life cycle.

rijalujjwal09@gmail.com

❑ Representing and Scheduling Project Plans

- A project manager has a wide variety of techniques available for depicting (representing) and documenting the project plans.
- These planning documents can take the graphical form or textual reports, although graphical reports have become the most popular means for representing the project plan.
- **The most commonly used methods are Gantt chart and Network Diagrams or Network Planning Models (CPM & PERT chart).**

❖ Gantt Chart

- Gantt charts are mainly used to allocate resources to the project activities. The resources allocated to the activities include staffs, hardware and software.
- Gantt charts are useful for resource planning and are a special type of bar chart where each bar represents an activity.
- The bars are drawn along the timelines. The length of each bar is proportional to the duration of the time planned for the corresponding activity.

rijalujjwal09@gmail.com

Contd. ….

Table: A sample of a Gantt chart

Devlopment Phase	4 Months						Duration
	0-20	20-40	40-60	60-80	80-100	100-120	
Requirement Gathering & Analysis							14
Designing							30
Coding							40
Testing							20
Implementation & Deployment							16
Total	14	30	40	20	16		120

rijalujjwal09@gmail.com

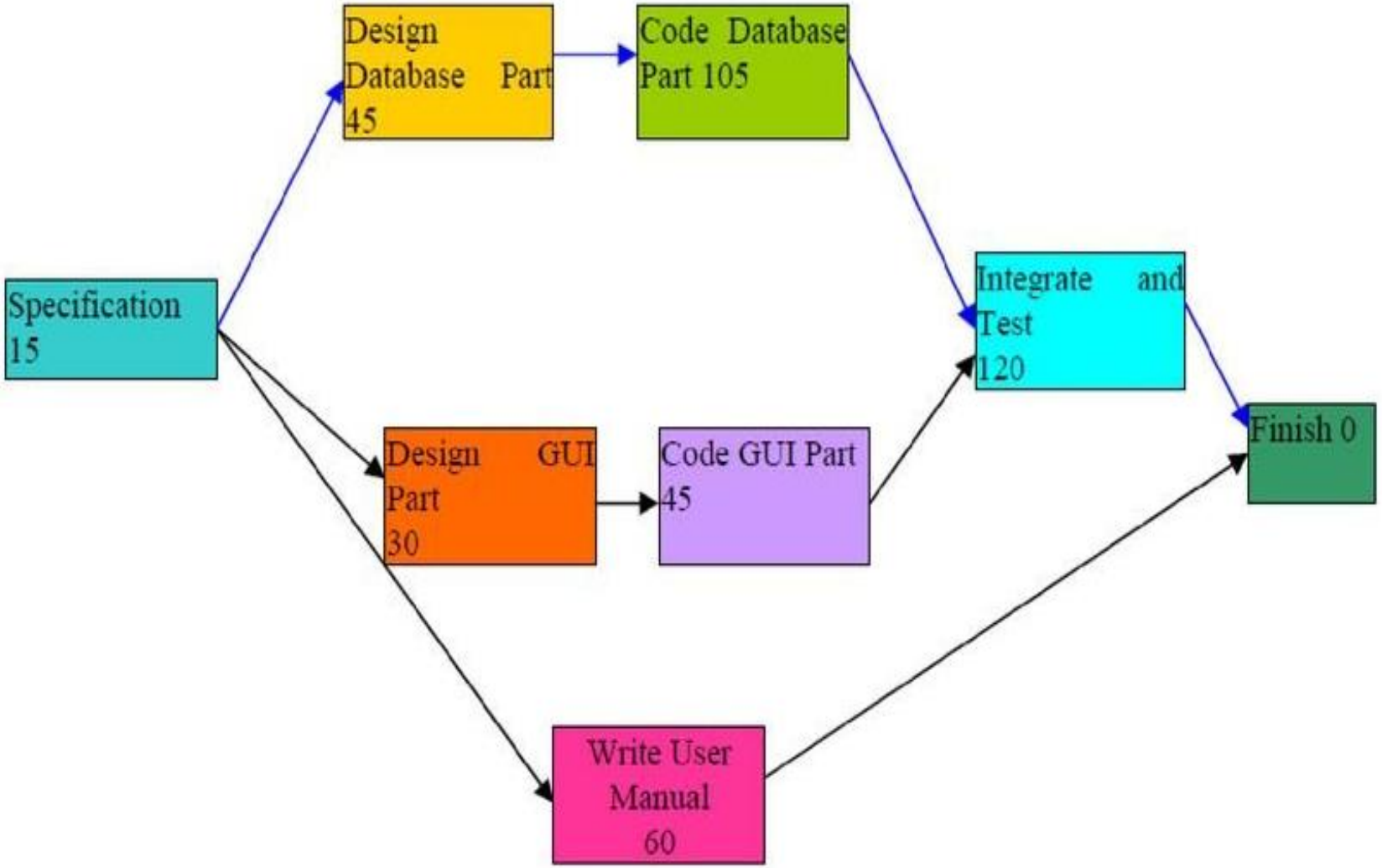
Contd. ….

❖ Critical Path Method (CPM)

- Work Break–down Structure (WBS) representation of the project is transformed into an activity network by representing the activities identified in WBS along with their inter-dependencies.
- An activity network shows the different activities making of a project, their estimated durations and inter-dependencies.
- Each activity is represented by a rectangular node and the duration of the activity is shown along-side.
- ***The figure below represents the activity network of an MIS software:***

rijalujjwal09@gmail.com

Contd. ….



rijalujjwal09@gmail.com

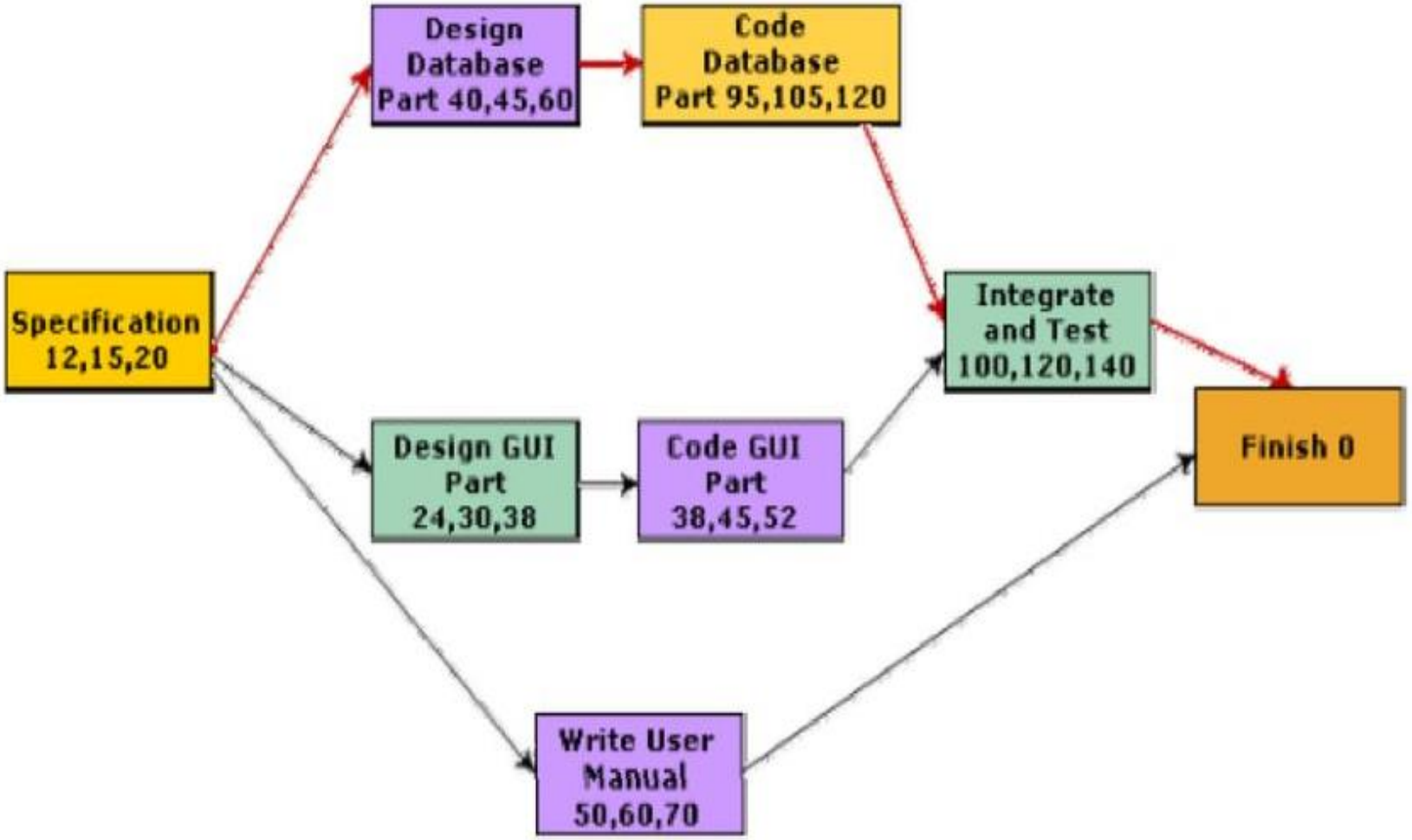
Contd. ….

❖ Program Evaluation and Review Technique (PERT) Chart

- PERT chart consists of a network of boxes and arrows. The boxes represent the activities and the arrows represent the task dependencies.
- The boxes of PERT charts are usually annotated with the pessimistic, likely, and optimistic estimates for every task.
- PERT chart is useful for monitoring the timely progress of the activities. Also, it is easier to identify all the parallel activities in a project using a PERT chart.
- ***The below figure shows the PERT chart of the MIS software problem:***

rijalujjwal09@gmail.com

Contd. ….



rijalujjwal09@gmail.com

❑ Differences between CPM and PERT Chart

Table: CPM vs. Gantt chart

Critical Path Method (CPM)	PERT Chart
i. Its origin is industry.	i. Its origin is military.
ii. Single estimate is considered for determining the duration of the activity.	ii. Three kinds of estimates are taken to determine the periods of activities; most likely, pessimistic, and optimistic.
iii. It does not consider uncertainty in time.	iii. It considers uncertainty in time.
iv. It is deterministic in nature.	iv. It is probabilistic in nature.
v. It does not use average time.	v. It uses average time.
vi. It is cost-based, etc.	vi. It is time-delayed for analysis of costs, etc.

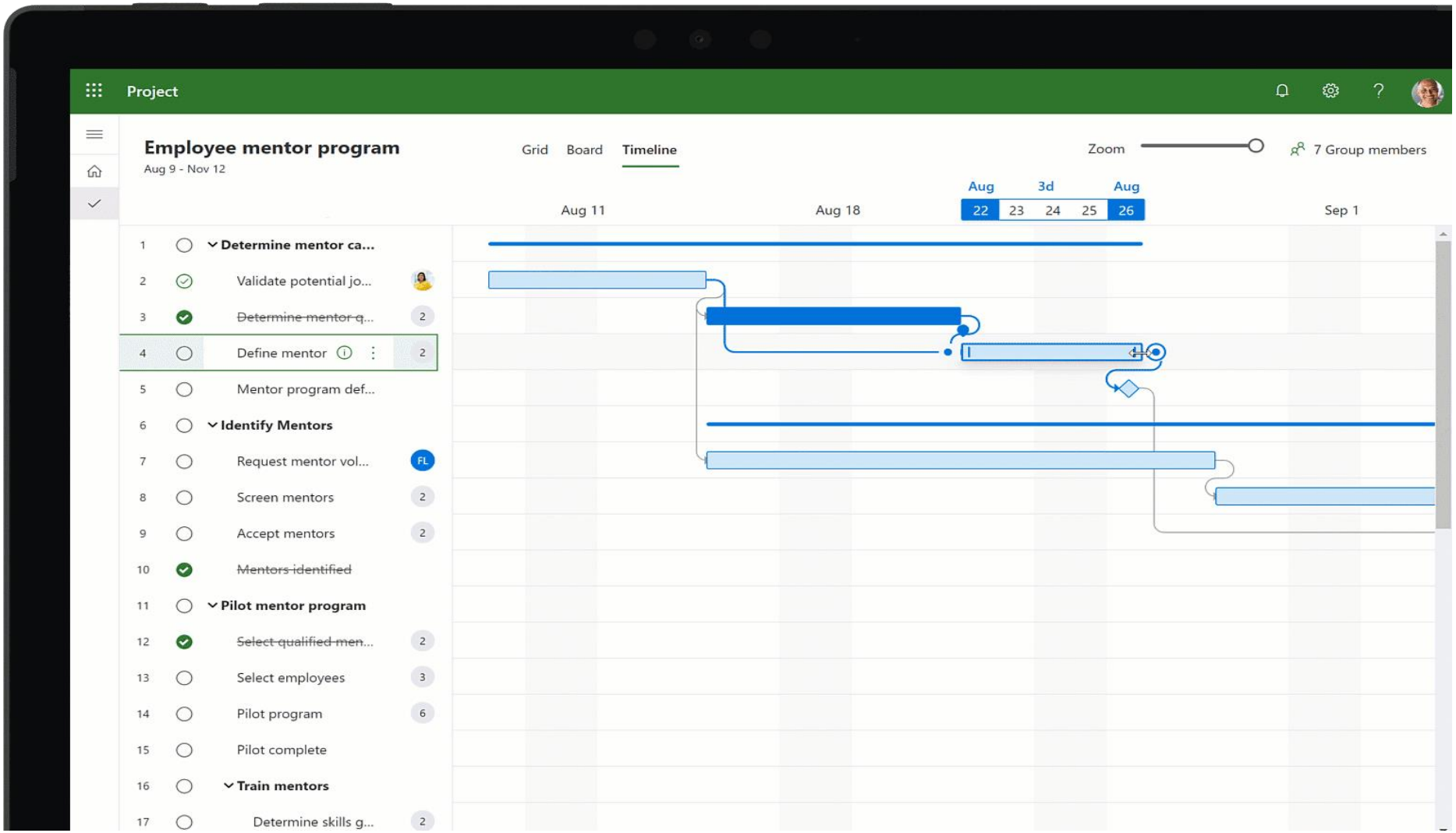
rijalujjwal09@gmail.com

❑ Using Project Management Software

- A wide variety of automated project management tools is available to help manage a development project.
- New versions of these tools are continuously being developed and released by software vendors.
- Project management tools are available to run on IBM – compatible personal computers, the Macintosh, and larger mainframe and workstation – based systems.
- Most programs offer features such as CPM/ PERT, Gantt charts, resource scheduling, project calendars, and cost tracking.
- As shown in the below figure, Microsoft Project is a full – featured program that holds the dominant share of the market.
- It is available as a software product for Windows and as an online service as part of Microsoft's Office 365.

rijalujjwal09@gmail.com

Contd. ….



rijalujjwal09@gmail.com